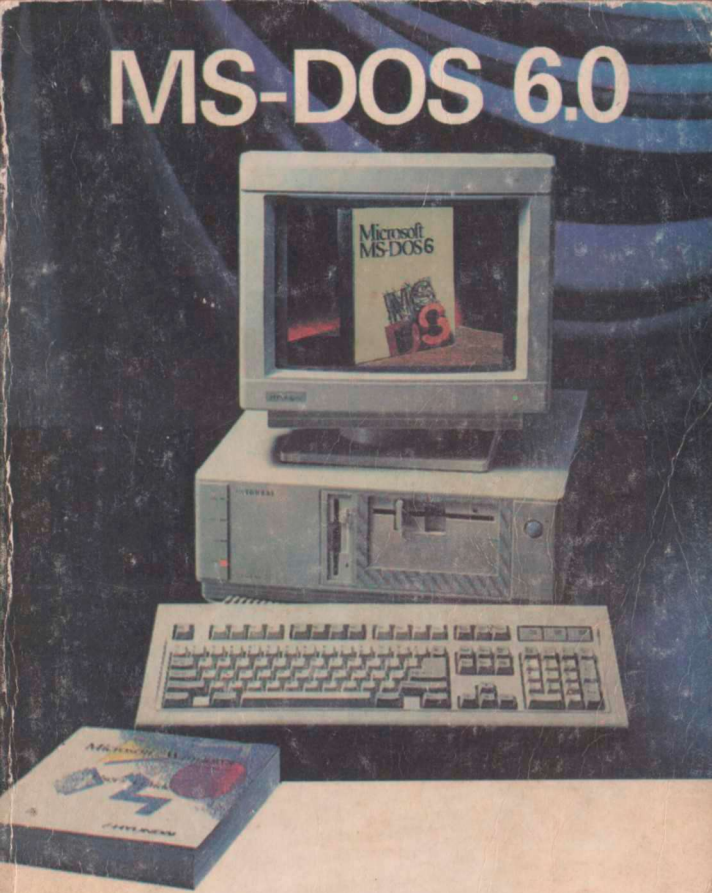
MS-DOS 6.0



Справочное руководство для пользователей компьютеров IBM-PC MS-DOS 6.0

MS-DOS 6.0

Справочное руководство
для пользователей
компьютеров IBM PC

Продукция фирмы "ВА Принт"

Москва
1994 г.

ББК 32.973.2-01я2
М93

"ВА ПРИНТ" предлагает

- *набор и редактирование текстов на персональном компьютере*
- *электронная верстка публикаций*
- *изготовление высококачественных оригинал-макетов*
- *художественное оформление изданий*
- *подготовка рекламных объявлений любого размера для периодической печати*

По Вашему желанию мы можем выполнить весь комплекс работ по выпуску художественной и специальной технической литературы.

Изготовим визитные карточки, конверты, бланки, буклеты, плакаты и другую продукцию.

Звоните:

в Москве:
(095) * 491 - 24 - 06,
499 - 23 - 28,
415 - 55 - 52

в Санкт - Петербурге:
(812) * 254 - 68 - 63, 567 - 52 - 26

в Ростове-на-Дону:
(8632) * 52 - 09 - 86

Следите за рекламой в новых изданиях - возможны изменения телефонов !

© Оформление "ВА ПРИНТ"
© Составление "ВА ПРИНТ"

ISBN 5-85734-023-3

© Авторское право и право на издание принадлежит фирме "ВА ПРИНТ"

По вопросам размещения рекламы обращайтесь по телефону: 491 - 24 - 06

1 Введение

С момента появления в 1981 году компьютера IBM PC и разработанной для него операционной системы MS-DOS фирмы Microsoft прошло уже более 10 лет. Но тем не менее ни одна другая операционная система по уровню своей популярности даже не может и соперничать с MS-DOS. Она "одушевляет" примерно 100 миллионов компьютеров по всему миру. Несмотря на то, что DOS (так мы будем для краткости называть MS-DOS) основана на разработках конца 70-х - начала 80-х годов, она является настоящим бестселлером среди операционных систем. Не так давно появились мощные операционные системы для персональных ЭВМ, такие как OS/2, версии UNIX для компьютеров IBM PC, появилась операционная оболочка Windows, а число проданных копий DOS продолжает расти.

Причиной этому является и огромное число программных продуктов, разработанных под DOS, и определенный консерватизм пользователей, и, что немаловажно, умеренные требования к аппаратным средствам. Последний из приведенных выше доводов в пользу DOS имеет особенно актуальное значение для нашей страны. По оценкам специалистов, 85% парка ПЭВМ у нас составляют машины класса IBM PC/AT, построенные на процессоре i286 с 1 MB оперативной памяти и винчестером на 40 MB. Естественно, что вычислительные возможности таких машин не позволят вам работать с новыми операционными системами и средами на более-менее приемлемой скорости. DOS же не требует больших системных ресурсов, она проста и надежна в работе.

Она одинаково хорошо будет работать и на простенькой по нынешним меркам IBM PC, и на мощной рабочей станции типа i486DX2/66. Кроме того, и это немаловажно, многие вещи DOS делает просто быстрее, чем другие операционные системы (ОС). Возьмите любую программу, ориентированную на текстовые режимы работы видеоподсистемы, например, любой текстовый редактор. Вы сразу почувствуете, что работа в DOS будет намного быстрее, чем в той же Windows.

Работа с САПР. Многие фирмы предпочитают эксплуатировать программное обеспечение САПР в среде DOS за ее быстродействие и из-за

того, что программы САПР очень часто используют нестандартные устройства ввода-вывода информации, с которыми работа в Windows и OS/2 невозможна. И таких примеров можно привести очень много. Несмотря на то, что в настоящее время количество программ, разрабатываемых для DOS, постепенно сокращается, DOS по оценкам ведущих западных специалистов в области компьютерных технологий на это десятилетие все равно останется доминирующей ОС для компьютеров типа IBM PC. И хотя фирма Microsoft прилагает сейчас значительные усилия для продвижения на рынок последней версии Windows 3.1, она же вынуждена развивать DOS, так как адекватной замены ей для более мощных машин так и не создано.

Наша книга призвана познакомить читателя с новейшей версией DOS - операционной системой MS-DOS 6.0, ее составом, командами и утилитами, а также с отличиями, появившимися в этом программном продукте по сравнению с предыдущими версиями DOS.

2 Организация компьютера IBM PC и требования MS-DOS 6.0 к ПЭВМ

С точки зрения рядового пользователя компьютера, IBM PC состоит из следующих основных узлов:

- *Центрального процессора (CPU)*, основного элемента ЭВМ, на который возложено выполнение всех арифметических, логических и других операций, а также функции управления всеми остальными устройствами компьютера. Как правило, машины IBM PC или совместимые с ними используют в качестве центральных процессоров процессоры фирмы Intel ряда 80x86 (8086, 80186, 80286, 80386, 80486 и Pentium), хотя ваш компьютер может быть оснащен процессором другой фирмы, например, AMD (Advanced Micro Devices).
- *Памяти*, выполняющей функции хранения как загруженных программ (оперативная память, или ОЗУ - RAM), так и специальных программ ОС (постоянная память, или ПЗУ - ROM).
- *Дисковой подсистемы*, на которую возложены функции хранения всех системных и прикладных программ. В качестве устройств, на которых осуществляется хранение данных, применяются чаще всего накопители на гибких (НГМД) и жестких (НМД) дисках. Однако ваша ПЭВМ может быть оснащена и другими устройствами для хранения информации, например, накопителями на магнитной ленте или оптических дисках (CD-ROM).
- *Видеоподсистемы*, которая осуществляет вывод графической и текстовой информации, получаемой в ходе работы вычислительной системы. В состав видеоподсистемы входят *видеомонитор*, обеспечивающий отображение информации и *видеоадаптер*, на который возложены функции преобразования сигналов, используемых для работы процессора, памяти, дисковой подсистемы и т.д. в сигналы, необходимые для работы видеомонитора. В настоящее время имеется много различных стандартов видеоподсистем, но наиболее рас-

пространенными считаются следующие: MDA, CGA, EGA, VGA и SuperVGA.

- *Клавиатуры*, с помощью которой осуществляется ввод информации в компьютер. Клавиатуры бывают различных типов, как по принципу действия, так и по количеству используемых клавиш и способу их размещения. В нашей стране наиболее распространены 101-клавишные клавиатуры, впервые разработанные фирмой IBM для своей машины IBM PC/AT.

Так как аббревиатура DOS переводится, как Дисковая Операционная Система (Disk Operation System), то компьютер, который работает с этой ОС, обязательно имеет в своем составе дисковые накопители, будь то самые первые машины IBM PC, имевшие только НГМД, или более поздние модели компьютеров, оснащенные НМД и другими устройствами хранения информации.

Типичная машина типа IBM PC оснащена, как правило, 2 НГМД и 1 НМД. Емкость дискет, используемых в PC, составляет от 180 K до 1.2 MB для дисководов 5.25" и от 720 K до 2.88 MB для дисководов 3.5". Емкость винчестера составляет 40 MB и более.

DOS организована таким образом, что логическим накопителям на НГМД присвоены буквы A и B. Если же ваш компьютер имеет 1 дисковод, то ОС будет обозначать его и как логический диск A, и как логический диск B. Обозначение логических дисков винчестера начинается с буквы C. Сам винчестер как физический диск может быть разбит на несколько логических дисков при помощи специальной программы FDISK, входящей в состав DOS. О работе этой программы мы расскажем далее. Как правило, один логический диск НМД используется для хранения самой ОС и небольшого количества сервисных программ. Другие же диски обычно используются для хранения всего остального программного обеспечения.

В качестве диска для хранения ОС чаще всего используется диск C, на котором размещаются файлы IO.SYS, MSDOS.SYS, COMMAND.COM, AUTOEXEC.BAT, CONFIG.SYS и отдельные программы операционной системы.

Для размещения MS-DOS 6.0 в полном объеме вам необходимо иметь примерно 6 MB дискового пространства, но если емкость вашего винчестера не позволяет выделить указанный объем дисковой памяти, вы можете установить усеченную версию DOS. Включите сюда наиболее необходимые вам программы, такие как FORMAT, CHKDSK, SYS, DISKCOPY а также файлы HIMEM.SYS, ANSY.SYS, RAMDRIVE.EXE и некоторые другие.

MS-DOS 6.0, как показывает ее эксплуатация авторами, успешно работает на машинах, оснащенных процессорами Intel 286, 386 и 486 с

объемом ОЗУ 640 К и более. Для своей работы она занимает чуть более 70 К оперативной памяти, что не намного более объема, занимавшегося предыдущей версией DOS 5.0.

Установив дополнительные драйверы, вы увеличите объем занимаемой оперативной памяти, но вместе с тем получите новые возможности, которые позволят вам повысить эффективность вашей вычислительной системы. Объемы памяти, занимаемые различными драйверами, будут приведены далее.

3 Состав DOS и ее основные модули

Операционная система представляет собой сложный программный комплекс. Им решаются следующие основные задачи:

- ОС осуществляет управление ходом вычислительного процесса.
- Предоставляет пользователю возможности общего управления вычислительной системой.

В рамках первой задачи ОС организует взаимодействие программ с внешними устройствами и между собой, управляет распределением оперативной памяти, выявляет различные события, возникающие в ходе работы компьютера, например, аварийные ситуации.

Общее управление машиной осуществляется с помощью командного языка, посредством которого пользователь осуществляет такие операции, как запуск программ, форматирование дисков, копирование файлов, установка режимов работы дисплея, печать информации на принтере и т.д.

Как уже было отмечено выше, ОС хранится на дисках в виде системных файлов, а после включения питания автоматически загружается в оперативную память компьютера, где и находится все время, пока компьютер находится в работе (или, говоря другими словами, является резидентной в памяти).

DOS может размещаться как на гибких (Floppy), так и на жестких (Hard) дисках, но, как правило, она размещается на жестких дисках, так как они обладают значительно большей емкостью, чем гибкие диски, и работают гораздо быстрее их.

Так как DOS довольно большая программа, то она выполнена в виде модулей. Это свойство, во-первых, позволяет изолировать друг от друга части ОС, облегчая ее разработку, во-вторых, дает возможность объединить в одном модуле взаимосвязанные логические функции.

DOS состоит из следующих основных модулей и программ:

- Базовой системы ввода-вывода (BIOS - Basic Input/Output System).
- Блока начальной загрузки (BOOT RECORD).
- Модуля расширения базовой системы ввода-вывода (IO.SYS).
- Модуля обработки прерываний (MSDOS.SYS).
- Командного процессора (COMMAND.COM).
- Модуля сжатия данных (DBLSPACE.BIN).
- Утилит DOS (FORMAT.COM, FDISK.EXE, LABEL.EXE и др.).
- Загружаемых драйверов устройств (HIMEM.SYS, RAMDRIVE.SYS и др.).
- Файлов AUTOEXEC.BAT и CONFIG.SYS.

Базовая система ввода-вывода (BIOS)

В архитектуре ПЭВМ BIOS занимает особое место. Ее можно рассматривать как составную часть аппаратных средств с одной стороны; с другой стороны, она является программным модулем DOS.

Что же представляет собой BIOS? Входящие в нее программы и данные осуществляют выполнение ряда важных функций по поддержке машины в работоспособном состоянии.

1. BIOS осуществляет тестирование основных компонентов машины после включения питания.
2. Вторая важная функция BIOS заключается в вызове блока начальной загрузки DOS. Поскольку, как отмечено выше, DOS - большая программа, состоящая из нескольких модулей, загрузка в память происходит в две ступени. Сначала BIOS загружает с системного диска в память блок начальной загрузки, а затем уже передает на него управление, и тот в свою очередь осуществляет загрузку всех остальных модулей DOS.
3. Еще одна важная функция BIOS заключается в обслуживании системных прерываний. Прерывания вырабатываются как аппаратными, так и программными средствами с целью выполнения различных операций.

Суть механизма прерываний заключается в том, что текущая работа машины, в чем бы она ни состояла, приостанавливается на короткий промежуток времени определенным сигналом, однозначно указывающим на возникновение ситуации, требующей немедленной обработки.

Блок начальной загрузки (Boot Record)

Блок начальной загрузки - это небольшая программа, единственное назначение которой заключается в считывании в оперативную память двух модулей DOS - IO.SYS и MSDOS.SYS. Сам блок начальной загрузки размещается всегда на одном и том же месте - на 0-й стороне, в 1-м секторе дорожки номер 00. Длина Boot Record составляет 512 байт, т.е. занимает всего 1 сектор.

Работа Boot Record заключается в следующем. Он просматривает каталог системного диска и осуществляет поиск файлов IO.SYS и MSDOS.SYS. Для упрощения и ускорения загрузки ОС эти файлы размещаются всегда первыми - один за другим последовательно. Если эти файлы не обнаружены, то диск считается несистемным, и система выдает соответствующее диагностическое сообщение. Если файлы ОС найдены, то производится их загрузка в память и передача им управления. Следует отметить, что Boot Record заносится на диск при форматировании всегда, независимо от того, является диск системным или нет.

Модуль расширения базовой системы ввода - вывода (IO.SYS)

Базовая система ввода-вывода является той частью ОС, которая менее всего подвержена изменениям. Расширение BIOS с помощью дополнительного модуля DOS придает ОС гибкость, позволяя управлять с ее помощью таким набором устройств, который бы наиболее соответствовал замыслу разработчиков ОС.

Этот модуль может быть изменен с учетом требований конкретной версии ОС. Использование механизма позволяет, с одной стороны, перекрывать в модуле расширения BIOS функции, обслуживаемые BIOS. С другой стороны, использование модуля расширения базовой системы ввода-вывода позволяет включать в BIOS новые внешние устройства (драйверы).

Драйверы позволяют поддерживать системой новые внешние устройства и организовать по-новому работу тех устройств компьютера в случаях, когда обмен информацией с ними должен происходить иначе, чем в стандартной версии DOS.

О необходимости подключения драйверов внешних устройств, а также об изменении других параметров DOS уведомляется через файл CONFIG.SYS. Какие команды включаются в файл CONFIG.SYS и каков порядок их использования будет рассказано в главе, посвященной файлам CONFIG.SYS и AUTOEXEC.BAT.

Кроме рассмотренных выше задач, модуль IO.SYS решает еще одну задачу. Он завершает загрузку в память ОС. С этой целью модуль расширения базовой системы ввода-вывода сначала передает управление на загруженный к этому моменту модуль обработки прерываний, в котором устанавливаются внутренние рабочие таблицы, инициируются вектора прерываний с номерами 20h-27h и производится подготовка к загрузке командного процессора, находящегося пока на системном диске. После этого управление снова возвращается модулю расширения BIOS. Модуль расширения BIOS производит загрузку командного процессора с диска в оперативную память и передает ему управление. На этом модуль расширения BIOS свою работу при запуске ОС завершает.

Модуль обработки прерываний (MSDOS.SYS)

В отличие от модуля IO.SYS, работающего с прерываниями нижнего уровня, (0h-20h) модуль MSDOS.SYS имеет дело с прерываниями верхнего уровня (21h-59h). Он образует верхний уровень системы, с которым взаимодействует абсолютное большинство программ.

В состав этого модуля входят подпрограммы, обеспечивающие работу файловой системы, устройств ввода-вывода информации, таких как клавиатура, дисплей, НГМД и НМД, принтер, последовательные порты. На него же возложена обработка специальных ситуаций, связанных с завершением программ, их преждевременным прерыванием и обработкой ошибок. В связи с исключительной важностью этого модуля для работы всей ОС его часто называют основным. Функции, реализуемые модулем MSDOS.SYS, в первую очередь используются самой операционной системой, например, команды DIR, COPY командного процессора, но с тем же успехом могут вызываться прикладными программами.

При обращении к функциям DOS из прикладных программ производится стандартный вызов прерывания 21h, при этом в один из регистров микропроцессора должен быть занесен номер вызываемой функции; в другие регистры вызывающая программа должна поместить аргументы выполняемой операции, если они необходимы, а по окончании выполнения прерывания из регистров могут быть получены результаты выполненной операции. Аналогичным образом производится обращение к прерываниям BIOS. С точки зрения программиста способы обращения к прерываниям BIOS и DOS совершенно идентичны. Однако работа с прерываниями BIOS дает большую близость к аппаратуре, избавляет от сервиса DOS, который иногда бывает избыточным. Следует также отметить, что при работе с BIOS, как правило, достигается большая

эффективность и быстродействие, однако прикладная программа при таком подходе оказывается в большей зависимости от конкретной модели ПЭВМ, чем при работе с прерываниями DOS.

Приведенный выше факт может сделать программу, написанную для одной ПЭВМ, несовместимой с другими типами компьютеров, однако в настоящее время острота этой проблемы снижена, так как различные фирмы-изготовители компьютерного оборудования стараются следовать общепринятым стандартам в написании BIOS.

Таким образом, выбор наиболее подходящего уровня и способа общения с аппаратными средствами определяется целями, критериями и сложностью разрабатываемой системы.

Командный процессор (COMMAND.COM)

Третий модуль DOS, располагающийся на системном диске - это *командный процессор*. В отличие от рассмотренных уже основных модулей DOS IO.SYS и MSDOS.SYS командный процессор может занимать на диске любое место и представляет собой не что иное, как обычную программу. Основные функции командного процессора заключаются в следующем:

- Прием и обработка команд, вводимых пользователем с клавиатуры или полученных из командного файла.
- Выполнение своих внутренних (встроенных) команд.
- Загрузка в память и исполнение внешних команд (утилит) DOS и прикладных программ. Прикладные программы и утилиты DOS, запускаемые на выполнение, должны иметь расширение .COM или .EXE. Запуск прикладных программ и утилит DOS абсолютно идентичен, так как и прикладные программы, и утилиты DOS организованы совершенно одинаковы. В общем случае можно утверждать, что для командного процессора не существует разницы между внешними командами DOS и прикладными программами.
- Вывод на экран монитора диагностических и аварийных сообщений и другие действия.

В функции командного процессора также входит выполнение файла автоматического запуска AUTOEXEC.BAT при загрузке машины.

Когда в качестве параметра командный процессор получает имя файла, не совпадающее с именами его внутренних команд, он производит анализ имени этого файла. Если файл имеет расширение .COM или .EXE, он считается загрузочным, если расширение файла имеет вид .BAT - он трактуется как командный файл. Исполняемые файлы в формате .COM фактически представляют собой дамп оперативной па-

мяти при работе файла типа .COM и не требуют настройки адресов после их загрузки в ОЗУ. Но размер файла типа .COM не может быть более одного сегмента памяти, то есть 64 К. Программы же типа .EXE требуют настройки по месту их размещения, то есть задания адресов соответствующих сегментов, но вместе тем на них не распространяется ограничение на длину программы. Размер программы в формате .EXE может превышать размер одного сегмента ОЗУ.

Если исполняемый файл является командным, то командный процессор начинает последовательно выполнять строки этого файла. Каждая строка командного файла представляет собой одну команду, метку или комментарий. Если в очередной строке стоит имя вызываемой программы, то COMMAND.COM передает ей управление, и начинается работа вызываемой программы. После ее завершения управление возвращается командному процессору.

При загрузке командного процессора в память он распадается на две части. Одна из них резидентна в памяти и находится в младших адресах памяти вслед за модулями расширения BIOS и модулем обработки прерываний. Нерезидентная часть располагается в старших адресах памяти. Резидентная часть содержит подпрограммы обработки прерываний 22h-24h. Эти прерывания прикладная программа может переопределить по-своему.

В резидентной части также располагается подпрограмма подзагрузки нерезидентной части командного процессора. Резидентная часть также включает в себя подпрограмму обработки файла AUTOEXEC.BAT при загрузке машины.

Нерезидентная часть COMMAND.COM содержит подпрограмму обработки команд DOS, поступающих с клавиатуры. В этой же части командного процессора располагается загрузчик внешних программ, который осуществляет чтение вызываемой программы с диска, настройку ее адресов, и передачу ей своего окружения (запуск вызванной программы на выполнение). Так как нерезидентная часть командного процессора располагается в старших адресах, любая программа, загружаемая в память, может затереть эту область памяти. После окончания работы такой программы управление всегда передается резидентной части командного процессора, которая пытается загрузить в память нерезидентную часть COMMAND.COM с диска.

Если ваша машина не имеет НМД или ваша система находится на гибком диске, то после окончания работы некоторых программ система может попросить вставить в дисковод дискету, содержащую командный процессор. Это сообщение говорит о том, что прикладная программа затерла нерезидентную часть командного процессора, а резидентная часть делает попытку загрузить нерезидентную часть.

Модуль сжатия данных (DBLSPACE.BIN)

Этот модуль является принципиально новым модулем, введенным в состав MS-DOS 6.0. Он выполняет сжатие данных при работе с дисковыми накопителями, что позволяет экономить дисковое пространство.

После включения питания компьютера DOS автоматически загружает этот модуль в память ПЭВМ вместе с другими модулями DOS. Однако до определенного момента этот модуль не оказывает никакого влияния на работу как самой DOS, так и программ. Для активизации этого модуля необходимо использование специального драйвера DBLSPACE.SYS, включенного в состав операционной системы. Подробно об использовании драйвера DBLSPACE.SYS рассказано в главе "Загружаемые драйверы устройств".

Утилиты DOS

В MS-DOS 6.0 входит более семи десятков утилит, предназначенных для выполнения различных функций. Однако в постоянной работе большинство утилит DOS пользователям ПЭВМ не требуются.

Наиболее часто используются такие программы, как FORMAT.COM, CHKDSK.EXE, SYS.COM, DISKCOPY.COM. Менее часто используются LABEL.EXE, TREE.COM, RESTORE.EXE, PRINT.EXE.

Важнейшее свойство DOS - любая программа может играть роль утилиты. Это делает DOS открытой для расширения. Неизменными являются только четыре модуля DOS: IO.SYS, MSDOS.SYS, COMMAND.COM и DBLSPACE.BIN. Вы можете переписать нужную вам внешнюю команду DOS на диск, и система будет работать с ней.

Загружаемые драйверы устройств

Загружаемые драйверы устройств позволяют, как отмечено выше, расширить функции стандартных устройств ПЭВМ или обеспечить управление дополнительными устройствами. Загружаемые драйверы устройств в ОС MS-DOS 6.0 имеют расширение .SYS или .EXE.

Наиболее широко используются такие драйверы, как HIMEM.SYS (управление дополнительной памятью - "Extended memory", или XMS), SMARTDRV.EXE (организация кэш-памяти для винчестера и гибких дисков), RAMDRIVE.SYS (организация электронного диска в оперативной памяти) и ANSI.SYS (расширенное управление клавиатурой и дисплеем). Реже применяются такие драйверы, как EMM386.EXE, DRIVER.SYS и другие.

Системное окружение DOS

В процессе своей загрузки DOS создает свое *окружение*, которое обычно называют *окружением командного процессора.*

Окружение DOS - это место в памяти, в котором она держит информацию, сообщаемую командами PATH, SET, PROMPT. Формат строк окружения приведен ниже и имеет следующий вид:

Переменная = значение_переменной

Каждая строка, задающая переменные окружения, заканчивается нулевым байтом. Операционная система MS-DOS организована таким образом, что существует ряд переменных системного окружения, имена которых известны и заранее определены.

Создаваемое в процессе начальной загрузки окружение содержит чаще всего в себе переменные COMSPEC, PROMPT и PATH, значения которых заносятся в системное окружение из файла AUTOEXEC.BAT автоматически. Примерный вид системного окружения приведен ниже:

```
comspec = c:\dos\command.com
prompt $p$g
path c:\;c:\dos;c:\nc;c:\nu;
```

Первая строка этого окружения описывает путь к файлу командного процессора, находящегося на диске C: в подкаталоге DOS. Указание пути к командному процессору необходимо для того, чтобы резидентная часть командного процессора смогла загрузить нерезидентную часть COMMAND.COM после затирания ее прикладными программами. Для того, чтобы установить эту строку в окружение командного процессора, необходимо включить в состав AUTOEXEC.BAT следующую команду

```
set comspec = c:\dos\command.com
```

Вторая строка в системном окружении задает вид системного запроса. Вид системного запроса может быть установлен командой PROMPT в файле AUTOEXEC.BAT. Если команда PROMPT не используется, то принимается вид системного запроса по умолчанию:

```
C:>
```

что довольно неудобно, особенно при работе с древовидной структурой организации файлов на дисках. Подробное описание команды PROMPT вы можете найти в разделе "Команды DOS".

Третья строка, имеющаяся в приведенном примере системного окружения, включает в себя переменную PATH, задающую пути альтерна-

тивного поиска файлов. Более подробно о назначении команды PATH рассказано в главе "Организация хранения данных в MS-DOS", раздел "Работа с древовидными каталогами". Полное описание команды PATH и примеры ее использования вы найдете в главе "Команды DOS".

Кроме вышеперечисленных переменных системного окружения пользователю предоставляется возможность самому задавать переменные системного окружения. Эта операция осуществляется с помощью команды DOS - SET. Обычно для задания своих переменных окружения эту команду включают в файл AUTOEXEC.BAT, что позволяет выполнять эту команду всякий раз при загрузке ОС.

Многие прикладные программы анализируют переменные системного окружения, ранее установленные командой SET. К их числу относятся такие широко известные программы, как:

- Текстовый редактор WORD - анализирует переменную TMP.
- Компоновщик объектных модулей LINK - анализирует переменную LIB.
- Операционная оболочка Windows - анализирует переменную TEMP, и многие другие программы.

Расскажем немного о том, как командный процессор осуществляет запуск других прикладных программ. После загрузки ОС активной программой является сам COMMAND.COM, который постоянно ожидает ввода с клавиатуры какой-нибудь команды. Получив какую-то команду, COMMAND.COM сначала проверяет, не является ли она его внутренней командой. Если полученная команда является его внутренней командой, то он выполняет ее, если нет то COMMAND.COM осуществляет поиск заданного имени файла. Найдя ее, он считывает программу в память и далее с помощью функции DOS - EXEC, передает ей свое окружение, активизируя эту программу.

Фактически передача окружения означает, что COMMAND.COM копирует свое окружение в новую область памяти, обычно перед загружаемой программой. Если загруженная программа в свою очередь активизирует еще одну программу (дочерний процесс) следующего уровня, то она также передает ей свое окружение. Таким образом, каждая программа, загруженная в память, имеет свою собственную копию окружения, причем копии программного окружения могут как совпадать, так и различаться. Родительский процесс (активная программа) может передать своему дочернему процессу свое окружение без изменения, а может и включить в него новую информацию. После завершения дочернего процесса управление от него возвращается в родительский процесс, а копия его программного окружения уничтожается.

Многие программы вообще не используют переменные окружения, и передача окружения им имеет чисто номинальный характер.

С другой стороны, передача окружения от родительских процессов дочерним является очень мощным средством передачи информации между процессами. Объем информации, передаваемой при использовании механизма передачи программного окружения, максимально составляет 32 K (32768 байт).

Файлы CONFIG.SYS и AUTOEXEC.BAT

Удобная рабочая обстановка для общения пользователя с машиной может быть создана при помощи двух специальных файлов:

- Файла конфигурации системы CONFIG.SYS.
- Файла автоматического запуска AUTOEXEC.BAT.

ФАЙЛ КОНФИГУРАЦИИ СИСТЕМЫ CONFIG.SYS

Этот файл по внешнему виду представляет собой обычный текстовый файл, состоящий из специальных директив конфигурирования DOS. С их помощью изменяются параметры установки DOS. Этот файл анализируется DOS при загрузке машины; если же он отсутствует, то DOS устанавливает конфигурацию по умолчанию, которая, впрочем, может оказаться неоптимальной для вашего персонального компьютера.

По сравнению с предыдущими версиями DOS файл CONFIG.SYS в версии MS-DOS 6.0 претерпел довольно значительные изменения. Появилась возможность настройки рабочей среды в нескольких вариантах использования системных ресурсов вашего компьютера. Кроме этого, появились новые директивы, используемые в файле конфигурации системы. Подробно используемые в CONFIG.SYS директивы будут рассмотрены в разделе "Директивы файла CONFIG.SYS".

Наиболее часто используемыми в файле CONFIG.SYS директивами являются следующие:

- *break* - для указания системе, как реагировать на нажатие комбинации клавиш Ctrl-Break или Ctrl-C (прерывание программы пользователем).
- *files* - для указания числа одновременно открытых файлов.
- *buffers* - указание числа буферов ввода-вывода для ускорения операций ввода-вывода.
- *shell* - для указания имени и местонахождения командного процессора.

Примерный вид файла CONFIG.SYS, созданного для MS-DOS 6.0 приведен ниже:

```
[menu]
menuitem = normal
menuitem = windows
menuitem = xms
menudefault = normal
[normal]
shell = c:\dos\command.com /e:1024 /p
device = c:\dos\himem.sys
dos = high,umb
device = c:\dos\ramdrive.sys 2048 512 /e
[windows]
shell = c:\dos\command.com /e:1024 /p
device = c:\dos\himem.sys
dos = high
device = c:\dos\smartdrv.exe 1024
stacks = 9,256
[xms]
shell = c:\dos\command.com /e:1024 /p
device = c:\dos\himem.sys
dos = high
device = c:\dos\ramdrive.sys 2048 512 /e
[common]
break = on
files = 60
buffers = 40
lastdrive = G
```

В процессе загрузки DOS 6.0 будет выполнены следующие действия. После появления сообщения "MS-DOS 6 Starting" на экране появится следующая картинка:

```
MS-DOS 6 Startup Menu

   1. NORMAL
   2. WINDOWS
   3. XMS

Enter a choice: 1



F5=Bypass startup files   F8=Confirm each CONFIG.SYS line [N]
```

Система просит ввести тот вариант работы, который необходим вам. Выберите его посредством либо клавиш Up, Down, либо нажав цифру, соответствующую варианту работы.

Если вы не хотите, чтобы файл CONFIG.SYS обрабатывался системой при загрузке, вы должны нажать клавишу F5 и затем Enter. Если же вы нажмете клавишу F8, то для выбранного варианта работы система будет обрабатывать строки файла CONFIG.SYS, спрашивая у вас, необходимо ли выполнять текущую строку файла конфигурации системы. После того, как вы закончите работу с меню запуска MS-DOS, система будет обрабатывать директивы CONFIG.SYS.

Директива [MENU] начинает собой описание пунктов меню запуска системы.

Директивы MENUITEM определяют пункты меню запуска системы: NORMAL, WINDOWS и XMS.

Директива MENUDEFAULT = NORMAL указывает системе, что вариантом работы по умолчанию является NORMAL.

Директива [NORMAL] начинает список директив, которые будут обработаны в варианте работы NORMAL. В результате выполнения этих директив системой будут выполнены следующие действия:

- Объявляется, что командный процессор находится на диске C: в каталоге \DOS и является резидентным в памяти. Размер его системного окружения увеличен до 1024 байт. Размер системного окружения обычно увеличивается для включения в его состав достаточно длинных строк в командах PATH, SET.
- Подключается драйвер дополнительной памяти.
- Объявляется, что DOS будет загружена в дополнительную память, и блоки старшей памяти (UMB) будут доступны как самой DOS, так и прикладным программам.
- Подключается драйвер электронного диска, который организует в дополнительной памяти диск размером 2 MB, с сектором 512 байт.

Аналогичным образом будут выполняться директивы, заключенные в других пунктах меню. Следует обратить особое внимание на директиву [COMMON], которая заключает в себя директивы, общие для всех пунктов меню запуска DOS.

В результате выполнения директив, входящих в [COMMON] ОС выполнит следующие действия:

- Будет разрешено прерывание программ пользователем при нажатии комбинации клавиш Ctrl-Break или Ctrl-C.
- Установится 60 буферов DOS для ускорения обмена данными с файлами.

- Будут выделено 40 дополнительных областей памяти для работы одновременно с 40 файлами.
- В системе возможно существование логических дисков от A: до G:.

ФАЙЛ АВТОМАТИЧЕСКОГО ЗАПУСКА AUTOEXEC.BAT

Файл автоматического запуска представляет собой самый обыкновенный пакетный (командный) файл, содержащий команды пакетной обработки. О файлах пакетной обработки мы расскажем в главе "Командные файлы". Однако AUTOEXEC.BAT играет особую роль в ОС MS-DOS. При загрузке DOS пытается найти этот файл в корневом каталоге системного диска после обработки файла CONFIG.SYS и автоматически начинает его выполнение без какого-либо побуждения пользователя ПЭВМ.

Обычно в файл AUTOEXEC.BAT вносятся команды, позволяющие произвести настройку системы и создать привычную рабочую обстановку. Рассмотрим для примера файл автоматического запуска для машины, используемой одним пользователем.

```
@echo off
verify = off
set comspec = c:\dos\command.com
goto %config%
:normal
path c:\;c:\dos;c:\nc;c:\nu;
goto end
:windows
path c:\;c:\dos;c:\windows;
goto end
:xms
path c:\;c:\dos;c:\nu;
goto endlast
:end
prompt $p$g
:endlast
c:\mouse\mouse
nc
```

Первая строка этого файла содержит команду пакетных файлов, подавляющую вывод на экран строк командного файла.

Далее следует команда, отключающая проверку правильности дисковых операций.

Далее команда SET указывает, откуда необходимо подгружать в память нерезидентную часть командного процессора. После этого опыт-

ные пользователи обнаружат строку %CONFIG%, являющуюся новой командой в файле AUTOEXEC.BAT по сравнению с предыдущими версиями DOS. Смысл ее появления в AUTOEXEC.BAT заключается в настройке различных режимов работы пользователя согласно выбранным в стартовом меню запуска DOS.

Для варианта работы NORMAL будут выполнены последовательно команды:

- PATH - указывает пути альтернативного поиска исполняемых файлов, что позволяет вызывать программы, содержащиеся в указанных в PATH с любого диска и каталога;
- команда GOTO END осуществляет переход на метку END; Выполнение строк файла AUTOEXEC.BAT для других режимов работы происходит аналогично. Далее для всех режимов работы системы будут выполнены строки после меток END или ENDLAST;
- команда PROMPT задает вид системного запроса, а затем происходит подключение драйвера манипулятора типа "мышь". В конце командного файла происходит запуск популярной программы - оболочки Norton Commander. Более подробно о командах, включаемых в командные файлы, и том, что это такое и для чего они нужны, вы узнаете в главе "Командные файлы".

Ввод информации с клавиатуры в MS-DOS

При работе на компьютере пользователь часто вводит с клавиатуры компьютера различные команды и другую информацию. Основным модулем, который организует обработку информации, поступающей с клавиатуры, в MS-DOS является *командный процессор* COMMAND.COM. Он организует выполнение команд, когда никакая другая программа не загружена в память.

При получении команд COMMAND.COM анализирует полученную команду на принадлежность ее к своим внутренним командам. Если команда является его встроенной командой, то COMMAND.COM выполняет ее, если команда не является встроенной командой командного процессора, то командный процессор запускает программу с указанным именем.

Работой клавиатуры управляет специализированный микропроцессор - *контроллер клавиатуры*, который воспринимает каждое нажатие клавиши и выдает ее код в специальный порт с адресом 60h.

Код нажатой клавиши, помещенный в порт с адресом 60h называется *скэн-кодом* (SCAN-CODE). Скэн-код всегда однозначно опеределяет

нажатую клавишу, но следует однако отметить, что каждой клавише как бы присваиваются два скэн-кода, отличающиеся на 80h. Один скэн-код с меньшим значением засылается при нажатии клавиши, второй - при ее отпускании. Описанная схема действует на всех компьютерах от IBM PC до IBM PC XT.

В машинах IBM PC AT, контроллер клавиатуры работает несколько по-иному. И при нажатии и при отпускании клавиш в выходной порт посылается один и тот же код, но при отпускании клавиши он предваряется кодом F0h. Хотя скэн-код и указывает однозначно на нажатую клавишу, по нему нельзя судить, включен ли на клавиатуре верхний или нижний регистр, или пользователь вводит русские или латинские буквы.

Кроме того, на клавиатуре компьютера имеются клавиши "Alt", "Ctrl", "Shift", "Caps Lock" и др. Таким образом, программы операционной системы должны анализировать не только скэн-код нажатой клавиши, но и анализ факта нажатия других клавиш "Shift", "Caps Lock" и др.

Организацией обработки нажатия клавиш в MS-DOS занимается специальная программа обработки прерываний клавиатуры.

Когда скэн-код клавиши выдается в порт 60h, процессор моментально прекращает свою работу и выполняет программу обработки прерываний - вызывается прерывание (INT 09h). Аббревиатурой INT обозначено программное прерывание DOS (от слова Interrupt - прерывание).

Помимо программных прерываний в компьютерах имеются также и аппаратные прерывания IRQ, которые выдаются устройствами компьютера.

Программа прерывания INT 09h, помимо порта с адресом 60h, работает также с двумя другими областями памяти: кольцевым буфером ввода, расположенным обычно в диапазоне адресов от 41Eh до 43Dh, куда помещаются ASCII коды нажатых клавиш, и областью памяти с адресом от 417h до 418h, хранящими слово состояния клавиатуры, в которой фиксируется нажатие клавиш ("Shift", "Caps Lock", "Num Lock", "Scroll Lock" и др.).

Значения битов в ячейках памяти 417h и 418h приведены в таблице 3.1 (для машин IBM PC AT).

Таблица 3.1

Ячейка с адресом	Бит	Клавиша	Бит установлен в 1
417h	7	Insert	режим вставки включен
	6	CapsLock	режим CapsLock включен
	5	NumLock	режим NumLock включен
	4	ScrollLock	режим ScrollLock включен
	3	Alt	клавиша нажата
	2	Ctrl	клавиша нажата
	1	левый Shift	клавиша нажата
	0	правый Shift	клавиша нажата
418h	7	Insert	клавиша нажата
	6	CapsLock	клавиша нажата
	5	NumLock	клавиша нажата
	4	ScrollLock	клавиша нажата
	3	Ctrl-NumLock	режим Ctrl-NumLock включен
	2	не использует.	
	1	не использует.	
	0	не использует.	

Прерывание клавиатуры немедленно обновляет биты этих ячеек памяти, как только будет нажата одна из клавиш-переключателей, даже если не было считано ни одного символа из буфера клавиатуры. Это верно и для клавиши "Ins", которая единственная из клавиш статуса клавиатуры помещает код в буфер (установка статуса "Ins" меняется, даже если в буфере нет места для символа).

Прерывание клавиатуры проверяет состояние статусных битов перед тем, как интерпретировать (преобразовать) скэн-код нажатой клавиши. При нажатии клавиши программа обработки прерываний считывает из порта 60h, скэн-код ее нажатия и, пользуясь имеющейся в ней таблицей трансляции скэн-кодов в коды ASCII, формирует двухбайтовый код, старший байт которого содержит скэн-код нажатой клавиши, а младший байт - код ASCII нажатой клавиши.

В процессе трансляции программа INT 09h анализирует статусные биты и в зависимости от их состояния формирует соответствующий код ASCII. Полученный в ходе трансляции двухбайтовый код засылается в буфер ввода, который служит для организации процесса взаимо-

действия DOS или прикладных программ с клавиатурой. Емкость буфера позволяет хранить 15 символов. Буфер организован по принципу FIFO (FIFO - First Input, First Output - первым пришел, первым вышел.)

Коды символов извлекаются из буфера в том порядке, в каком они в него поступают. В том случае, если буфер переполнен, поступление новых символов блокируется, а пользователю подается звуковой сигнал. Когда компьютер находится в состоянии ожидания, управление вводом символов в компьютер осуществляет командный процессор COMMAND.COM.

При появлении символа в командной строке COMMAND.COM при помощи соответствующих программ DOS переносит его в свой буфер, очищая при этом кольцевой буфер ввода для клавиатуры, а также выводит символ на экран в режиме эхо-повтора. При получении кода клавиши Enter (0Dh) командный процессор предполагает, что ввод команды завершен, и анализирует содержимое своего буфера по описанной выше схеме. При этом следует отметить, что командный процессор работает только с кодами ASCII.

Если же компьютер выполняет какую-либо программу, и при этом с клавиатуры поступает информация, то программа сама организует ввод данных с клавиатуры и вывод их на дисплей. Чаще всего прикладные программы для этих целей используют стандартные средства DOS, но могут включать и свои программы обработки прерываний. В процессе работы с прикладной программой пользователь может нажать какие-либо символы, ввод которых не требуется прикладной программе. В этом случае после завершения программы и передаче управления COMMAND.COM командный процессор проанализирует содержимое кольцевого буфера клавиатуры и поместит их в свой буфер.

Кроме клавиш, обозначающих символы алфавита, цифры, знаки препинания и специальные знаки, на клавиатуре имеются клавиши, которые не отображаются на экране. К таким клавишам относится группа функциональных клавиш: "F1...F10 (F11,F12 для IBM PC AT)", клавиши "Home", "End", "↑", "↓" и др. Для этих клавиш также существуют свои скэн-коды, которые, однако, транслируются в коды ASCII несколько по-иному, чем рассмотренные символьные, цифровые клавиши, клавиши знаков препинания и специальных символов. Таким клавишам программа обработки прерывания от клавиатуры INT 09h назначает двухбайтовый код, в котором младшему байту (ASCII-код клавиши) присвоено значение 0.

Двухбайтовые коды, содержащие на месте ASCII-кода ноль, называются *расширенными ASCII-кодами*. Эти коды находят широкое применение в утилитах DOS и прикладных программах. Например, в оболочке

DOS-программе MS-DOS Shell клавиша F5 используется для обновления содержимого экрана MS-DOS Shell (функция "Refresh" пункта "View" главного меню программы), а комбинация клавиш Shift-F5 - для перезакраски экрана оболочки (функция "Repaint Screen" пункта "View").

Как свидетельствует эволюция развития персональных компьютеров и программного обеспечения, на каком-то этапе их развития потребовались дополнительные возможности ввода с компьютера информации, отличной от обычного текста. Для решения этой проблемы в компьютерах IBM PC появилась возможность вводить коды не только функциональных клавиш и клавиш управления курсором, но и коды комбинаций этих клавиш вместе с клавишами "Alt", "Ctrl", "Shift".

Комбинации функциональных клавиш и клавиш управления курсором, вместе с клавишами "Shift", "Alt", "Ctrl" также образуют расширенные коды. Однако в этом случае в старший байт расширенного кода ASCII помещается не скэн-код нажатой клавиши, а некоторый специальный код, назначенный комбинации клавиш. Подробную информацию о скэн-кодах клавиш можно найти в главе "Приложения" (см. Приложение 4).

4 Организация хранения данных в MS-DOS

Файлы

Чтобы программа могла обратиться к своим данным, должны существовать какие-то средства, позволяющие упорядочить размещение данных, обеспечить быстрый и надежный доступ к информации.

Совокупность средств, обеспечивающих доступ к данным, называется *системой управления файлами*, или *файловой системой*.

Файлом называется любая последовательность данных определенной длины, имеющая имя.

Файл может хранить текст, программу, числовые данные, закодированные графические изображения. Длина файлов, как и все, что связано с памятью, измеряется в байтах.

Имена файлов в операционной системе MS-DOS составляются по определенным правилам:

- Имя должно содержать, как минимум один, но не более 8 символов.
- Имя файла может иметь расширение, которое отделяется от имени файла точкой и должно содержать не более 3 символов.
- Для записи имен файлов и расширений используются буквы латинского алфавита (A-Z), цифры (0-9) и некоторые специальные знаки: -, _, $, &, @, %, (,), ^, ', `, {, }, ~, !, #.
- В качестве имен файлов запрещено использование словосочетаний, обозначающих имена стандартных устройств DOS. Обозначение стандартных устройств DOS приведено в таблице 4.1.

Таблица 4.1

Обозначения стандартных устройств DOS	
Обозначение устройства	Его функции
AUX	1 коммуникационный порт
CON	Консоль (дисплей и клавиатура)
PRN	1 параллельный порт
NUL	Пустое устройство (нуль - устройство)
CLOCK$	Системные часы
COM1	Синоним AUX
COM2	2 коммуникационный порт
LPT1	Синоним PRN
LPT2	2 параллельный порт
LPT3	3 параллельный порт

Имя файла обычно составляется так, чтобы пользователь мог легко вспомнить, какая информация хранится в файле. Расширение имени обычно указывает на тип хранящихся в нем данным. Как уже было отмечено выше расширения .COM и .EXE указывают на то, что файл является выполняемым. Расширение .TXT указывает на то, что файл хранит текстовую информацию, а расширение .BAS содержит программу на языке программирования Бэйсик. Стандартные расширения файлов, принятые в ОС MS-DOS, приведены в таблице 4.2.

Таблица 4.2

Стандартные расширения имен файлов, принятые в ОС MS-DOS	
Расширение	Примечание
.ASM	исходный текст программы на языке Ассемблер
.BAK	файл с предыдущим вариантом файла (резервный)
.BAS	исходный текст программы на языке Бэйсик
.BIN	двоичный программный файл
.C	исходный текст программы на языке Си
.COM	выполняемый файл
.CPI	файл с данными кодовой страницы

Стандартные расширения имен файлов, принятые в ОС MS-DOS	
.DAT	файл данных
.DOC	файл с документацией
.EXE	выполняемый файл
.HLP	файл помощи
.LST	файл с листингом программы
.LIB	библиотечный файл
.MAP	листинг компоновки выполнимого модуля
.OBJ	объектный файл программы
.OVL	оверлейный файл
.OVR	оверлейный файл
.PAS	исходный текст программы на языке Паскаль
.SYS	драйвер устройства
.TMP	временный файл
.TXT	текстовый файл
.$xx	временный или ошибочный файл

Типовые расширения имен очень удобны, так как позволяют обращаться в случаях файлов с расширениями типа .COM, .EXE и .BAT, задавая только имя файла. Если же вам необходимо выбрать с диска все файлы определенного типа, вы можете только указать расширение и выдать их на экран дисплея или принтер.

Для обращения к группам файлов применяются *групповые шаблоны*, образуемые с помощью символов * и ?. Символ "*", встречающийся в имени файла, трактуется DOS как "любая последовательность символов", а символ "?" - как "любой один символ". Ниже приведены примеры использования групповых шаблонов и необходимые пояснения к ним:

- *.EXE - все файлы с расширением имени .EXE.
- A*.COM - все файлы типа .COM, имена которых начинаются с буквы A.
- *.* - все файлы текущего каталога.
- ???.BAS - все файлы с расширением .BAS и с именами, имеющими не более трех букв.
- A?.* - все файлы, имена которых начинаются с A и состоят из одной или двух букв.

Групповые шаблоны удобно указывать в командах COPY, DIR, XCOPY, REN и других. Например, вы хотите получить сведения о всех файлах, имеющих расширение .TXT.

После ввода команды

```
A> dir *.txt
```

DOS выведет на экран дисплея список хранящихся на диске A: файлов с расширением .TXT. При необходимости скопировать с диска B: все файлы на диск C: в подкаталог PRIMER следует ввести команду

```
B> copy *.* c:\primer
```

Результатом выполнения этой команды станет копирование всех файлов с диска A: на диск C: в подкаталог PRIMER.

К имени файла может быть добавлена приставка, указывающая дисковое запоминающее устройство, на котором хранится или будет храниться файл. Примеры использования приставок с именами файлов приведены ниже.

```
a:notes.txt
c:program.exe
```

Имя файла с приставкой называется *спецификацией файла*. Кроме идентификатора диска, спецификация файла может также включать дополнительные указатели о местонахождении файла на диске с древовидной структурой каталогов.

Об особенностях составления и работы с полной спецификацией файлов подробно будет рассказано в разделе "Работа с древовидными каталогами".

Файловая структура диска

Для того, чтобы обратиться к данным, расположенным на диске, необходимо знать адрес первого сектора из тех, в которых хранятся данные файла. Адрес любого сектора определяется следующими координатами:

- Номером дорожки (цилиндра).
- Номером поверхности.
- Номером сегмента.

Пользоваться адресом сектора для записи или чтения информации можно, но практически такой способ используется очень редко, так как он довольно трудоемок. Однако могут возникнуть ситуации, когда вам необходимо будет непосредственно обратиться к какому-либо сектору

диска, поэтому мы считаем необходимым рассказать о том, что представляет собой файловая структура в MS-DOS и как она организована. Следует отметить, что для выполнения таких специфических работ имеются специальные средства, например, программа Disk Editor из инструментального пакета Norton Utilites.

Диски одного компьютера могут хранить сотни и тысячи файлов, и для обеспечения возможности обращения к ним пришлось бы вести и составлять громоздкие каталоги, в которых бы для каждого файла были указаны занимаемые им сектора. Однако операционная система освобождает пользователей от необходимости вести подобные каталоги и делает это сама, автоматически.

В общем случае схема обращения к данным файла такова. Пользователь обращается к файлу, указывая операционной системе его имя или его спецификацию. Операционная система, прежде чем выполнить указанные пользователем действия, обращается к своей системной области (файлу - каталогу), находит в нем запись, соответствующую указанному имени файла, определяет местоположение секторов, занимаемых файлом, и только после этого выполняет заданные пользователем действия.

В процессе доступа к данным система использует также хранящуюся на диске специальную таблицу. Эта таблица называется *таблицей размещения файлов FAT (File Allocation Table)*.

Для работы также используется *стартовый сектор*. О назначении стартового сектора и FAT, а также функциях, которые возложены на них, будет рассказано далее.

Стартовый сектор, каталог, таблица размещения файлов и оставшееся свободное пространство диска, называемое *областью данных*, являются элементами файловой структуры диска, создаваемыми ОС в процессе подготовки диска к работе.

Операция подготовки диска к работе называется *инициализацией*. Она выполняется с диском перед тем, как начать его использовать. Если вы подвергнете инициализации диск, уже бывший в работе, то вся информация будет на нем потеряна. Впрочем, существуют ситуации, когда информацию на диске можно восстановить и после его инициализации. Вопрос восстановления информации на диске после его инициализации будет подробно рассмотрен в главе "Команды MS-DOS" (см. команду UNFORMAT).

Объемы, занимаемые на дисках элементами файловой структуры, различаются в зависимости от формата диска. Для гибких дисков соответствующие отношения приведены в таблице 4.3.

Таблица 4.3

Объемы, занимаемые элементами файловой структуры на гибких дисках

Формат диска	Общее кол-во секторов	Объем системных областей				Область данных (секторов)
		всего	в том числе			
			старт. сект.	FAT	каталог	
S-8 (5.25")	320	7	1	2	4	313
D-8 (5.25")	640	10	1	2	7	630
S-9 (5.25")	360	9	1	4	4	351
D-9 (5.25")	720	12	1	4	7	708
QD-9 (3.5 ")	1440	18	1	10	7	1422
QD-15 (5.25")	2400	29	1	14	14	2371
QD-18 (3.5 ")	2880	29	1	14	14	2851
QD-36 (3.5 ")	5760	-	1	-	-	-

На жестких дисках нет четко определенных объемов дисковой памяти, занимаемых элементами файловой структуры. Мы уже упоминали, что жесткий диск может быть разбит на несколько логических дисков, каждый из них используется автономно. Так как размеры логических дисков винчестера могут быть различными, то при инициализации DOS сама определит объемы дисковой памяти, которые будут занимать элементы файловой структуры.

Стартовый сектор - это своего рода визитная карточка любого диска. Он содержит необходимые ОС сведения для работы с диском. Под него выделяется самый первый сектор диска на дорожке (цилиндре) 00 поверхности 0. В него записываются следующие характеристики:

- Идентификатор системы, если на диске записана ОС.
- Размер секторов диска в байтах.
- Количество копий FAT (обычно имеется 2 копии FAT).
- Количество элементов в корневом каталоге.
- Количество секторов на диске.
- Указатель типа магнитного носителя информации.
- Количество секторов, занимаемых FAT.
- Количество секторов на дорожку.
- Количество поверхностей.

Далее стартовый сектор содержит собственно программу блока начальной загрузки DOS, которая используется только на загружаемых дисках для загрузки операционной системы после включения питания. За стартовым сектором на диске следует таблица размещения файлов.

ТАБЛИЦА РАЗМЕЩЕНИЯ ФАЙЛОВ (FAT)

Таблица размещения файлов является важнейшей файловой структурой диска. В процессе работы содержимое области данных постоянно меняется: добавляются новые файлы, изменяется их содержимое, файлы стираются с диска и т.д. Выполнение этих операций требует наличия специального механизма, позволяющего распределять дисковое пространство между файлами и обеспечивать доступ к ним. Механизмом, реализующим распределение дискового пространства доступа к файлам в DOS и выступает таблица размещения файлов.

При выполнение операций чтения-записи информации обмен между дисковыми накопителями и памятью компьютера осуществляется блоками. Минимальный объем блока равен 1 сектору на диске. Для уменьшения количества обращений к дискам в DOS применен следующий подход. Информация за одно обращение к диску может быть считана или записана из нескольких последовательно расположенных секторов, образующих особый логический блок, называемый *кластером*. Размеры кластеров зависят от емкости используемого диска. Для гибких дисков емкостью 1.2 и 1.44 MB кластеры включают в себя 1 сектор, а на жестких дисках в зависимости от их емкости кластеры могут составлять 4 или 8 секторов.

Файлу, записываемому на диск, выделяется *целое* число кластеров, причем, что существенно, кластеры могут находиться в различных местах диска. В отличие от файлов, хранящихся в непрерывных областях памяти (*непрерывные файлы*), файлы, хранящиеся в различных кластерах, называются *фрагментированными*.

Назначение FAT заключается в хранении на дисках фрагментов файлов. К сожалению, сама DOS не позволяет исключить фрагментацию файлов, и поэтому были разработаны специальные программы-дефрагментаторы дисков. Наиболее популярной из них является Speed Disk из пакета Norton Utilites. Впечатляющий успех программ-дефрагментаторов побудил разработчиков DOS включить в версию MS-DOS 6.0 свою программу-дефрагментатор диска DEFRAG. Об этой программе вы узнаете, прочитав главу "Команды MS-DOS".

Механизм доступа к файлам реализован в DOS с использованием FAT следующим образом. Область данных диска рассматривается как последовательность пронумерованных кластеров. Каждому кластеру в FAT ставится в соответствие элемент FAT с тем же номером. Например, элемент 100 FAT соответствует 100-му кластеру диска, а элемент 101 соответствует 101-му кластеру диска. В каталоге, содержащем сведения о файлах на диске, для каждого файла указан номер первого кластера, занимаемого файлом. Этот номер называется *точкой входа* в FAT. Система для того, чтобы, например, прочитать данные о файле,

обращается в каталог, находит номер соответствующий имени файла и затем обращается в FAT. Войдя в FAT по точке входа, ОС читает соответствующую ячейку FAT, в которой записан номер следующего занимаемого файлом кластера. Далее ОС снова обращается в FAT, где читает номер следующего кластера и так до тех пор, пока ячейка FAT не будет содержать специального кода FFF, говорящего системе, что этот кластер - последний кластер файла.

Пример на рисунках 4.1 и 4.2 иллюстрирует процесс обращения к файлу, занимающему 5 кластеров на диске.

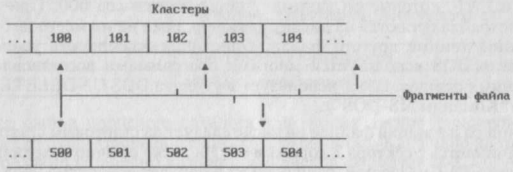


Рис 4.1

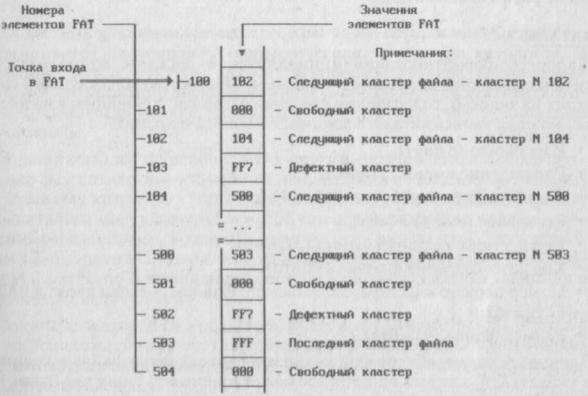


Рис. 4.2

Как видно из рисунка, первый фрагмент файла занимает кластеры номер 100, 102, 104, а второй фрагмент - кластеры 500 и 503. Кластеры 103 и 502 являются дефектными, они отмечены специальным кодом FF7. Этот код говорит о том, что в процессе форматирования (инициализации) диска система обнаружила на этом участке диска дефекты поверхности магнитного носителя, препятствующие нормальному считыванию-записи информации.

Кластеры 101, 501 и 504 имеют коды 000, т.е. они свободны. Кластер 503 имеет код FFF, который указывает ОС на то, что этот кластер является последним кластером файла. Если файл удаляется, то в те ячейки FAT, которые он занимал, записывается код 000. При этом данные файла остаются на диске до тех пор, пока на их место не будут записаны данные другого файла. Описанная особенность удаления файлов в DOS используется многими программами восстановления удаленных файлов. Ее же использует и утилита DOS UNDELETE (см. главу "Команды MS-DOS").

Таблица размещения файлов на диске следует за стартовым сектором, т.е. начинается с сектора 2 дорожки 00. На диске обычно сохраняются две копии FAT на случай, если одна из них будет испорчена. Под FAT отводится количество секторов, указанное в таблице 4.1 для гибких дисков и умноженное на число копий FAT. Вслед за копиями FAT на диске расположен каталог.

КАТАЛОГИ

В процессе форматирования (инициализации) диска ОС всегда создает каталог, который называется корневым. Файл корневого каталога состоит из записей, содержащих сведения о файлах, хранящихся на диске. Каждая запись каталога занимает 32 байта и хранит:

- Имя файла (8 байт).
- Расширение имени (3 байта).
- Двоичный код атрибутов файла (1 байт).
- Резервное поле (10 байт).
- Код времени создания файла (2 байта).
- Код даты создания файла (2 байта).
- Номер первого кластера, занимаемого файлом, - точка входа в FAT (2 байта).
- Размер файла (4 байта).

Двоичный код атрибутов файла характеризует статус файла и хранит 6 указателей, каждый из которых может принимать значение 0 или 1. Значения указателей байта атрибутов приведены в таблице 4.4.

Таблица 4.4.

Значения указателей байта атрибутов		
Указатель	Бит	Значение
0	1	Read-only (файл только для чтения)
1	1	Hidden (файл скрытый - не показывается командой DIR)
2	1	System (файл системный (файлы IO.SYS и MSDOS.SYS)
3	1	Метка тома, такая запись существует только в корневом каталоге
4	1	Файл является подкаталогом
5	1	Файл является архивными

Размер файла корневого каталога для гибких дисков фиксирован и зависит от формата физической структуры диска. Как видно из таблицы, под файл корневого каталога для дисков форматов D-8, QD-9 под файл каталога отведено 7 секторов по 512 байт. Корневой каталог этих дисков может хранить сведения о 112 файлах. Этого количества обычно достаточно для гибких дисков, которые не имеют развитой структуры каталогов.

Для доступа к данным винчестера, хранящим сотни и тысячи файлов, применяется древовидная (иерархическая) структура каталогов. В вершине этой структуры находится корневой каталог, создаваемый при форматировании диска. Ветви дерева образуют подкаталоги, которые создают или удаляют либо сами пользователи, либо прикладные программы.

Старший каталог по отношению к другим, подчиненным непосредственно ему каталогам, называется *родительским каталогом*, а подчиненные ему каталоги - "подкаталогами 1-го уровня". В свою очередь, подкаталоги могут содержать в себе как обычные файлы, так и подчиненные им каталоги, которые в свою очередь называются "подкаталогами 2-го уровня". Положение на диске корневого каталога фиксировано и известно операционной системе - он располагается, начиная с секторов, следующих за FAT.

Подкаталоги хранятся в области данных как обычные файлы. Данные о местонахождении подкаталогов содержит корневой каталог. Подкаталоги корневого каталога хранят сведения о подчиненных им подкаталогах и файлах.

Отличие подкаталога от обычного файла в том, что бит 4 байта атрибутов файла установлен в 1, и поле размера файла имеет значение 0.

Размер файла подкаталога неограничен, что позволяет хранить в подкаталоге неограниченное количество файлов в пределах емкости диска. Первый и второй элементы в поле "Имя файла" имеют значения соответственно "." и "..".

Элемент с именем "." описывает собственно подкаталог, а элемент с именем ".." - родительский каталог. Наличие в подкаталоге элементов с данными о родительском каталоге позволяет пользователям продвигаться по дереву каталогов не только от корневого каталога к подчиненным, но и в обратном направлении.

ОБЛАСТЬ ДАННЫХ

Область данных располагается вслед за корневым каталогом и занимает все остальное пространство на диске. Файлам, хранящимся на диске, по возможности предоставляется непрерывная последовательность кластеров.

Если файл расширяется или записывается на место ранее удаленного более короткого файла, то файлу может быть выделено несколько областей памяти на диске - диск становится фрагментированным. Связь между отдельными фрагментами файлов обеспечивается посредством указателей FAT. Со временем степень фрагментации, как было отмечено выше, возрастает. Но указанное обстоятельство не влияет на емкость вашего винчестера. Однако время доступа к данным файла может существенно увеличиться, потому что для записи или чтения информации головкам записи-чтения придется перемещаться столько раз, сколько фрагментов содержит файл. Если же файл непрерывен, то головки записи-чтения необходимо будет подвести только один раз.

Упростить фрагментацию файлов можно, если использовать специальные утилиты дефрагментации, например уже упоминавшийся дефрагментатор DEFRAG, входящий в состав MS-DOS 6.0.

Суть операции дефрагментации состоит практически в переписывании содержимого диска, но каждый из хранящихся файлов вновь записывается на диск как непрерывный.

РАБОТА С ДРЕВОВИДНЫМИ КАТАЛОГАМИ

На рисунке 4.3 приведен пример каталога, который иллюстрирует примерный вид типичной древовидной структуры файлов, хотя следует оговориться, что эта структура весьма условна, так как организация подкаталогов на дисках может быть очень разветвленной.

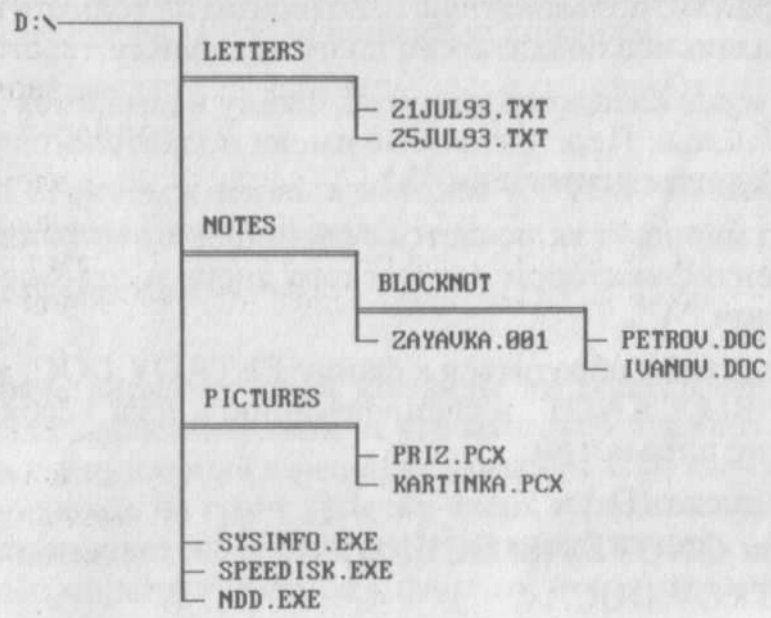


Рис. 4.3

В вершине дерева каталогов находится *корневой каталог*, создаваемый ОС в процессе форматирования диска и не имеющий собственного имени. Он обозначен на схеме символом "\". Корневой каталог включает в себя как три подкаталога первого уровня LETTERS, NOTES и PICTURES, но и три обыкновенных файла - программы SYSINSFO, SPEEDISK и Norton Disk Doctor (файл NDD) из пакета Norton Utilites.

Имена подкаталогам присваиваются при их создании так же, как и обычным файлам, к имени подкаталога можно добавить и расширение, но, как правило, этой возможностью пользуются редко. Подкаталог LETTERS хранит файлы писем 21JUL93.TXT и 25JUL93.TXT.

Аналогичным образом организовано хранение файлов и в каталоге PICTURES, в котором расположены файлы графических изображений (PRIZ.PCX, KARTINKA.PCX) в широко распространенном графическом формате PCX. Подкаталог же NOTES по сравнению с ранее рассмотренными обладает особенностями. Он хранит как обычный текстовый файл ZAYAVKA.001, но и кроме этого имеет подчиненный подкаталог BLOCKNOT, содержащий в себе еще 2 файла PETROV.DOC и IVANOV.DOC.

ПЕРЕМЕЩЕНИЕ ПО ПОДКАТАЛОГАМ

Для того, чтобы системе провести операцию чтения или записи файлов, в древовидной структуре каталогов системе необходимо просмотреть всю цепочку файлов на пути от корневого каталога до подкаталога, в котором хранится файл. Таким образом, для того, чтобы ОС могла

получить доступ к файлу, пользователю необходимо не только указать имя файла, но и указать все подкаталоги на пути к файлу.

Указатель перечня имен каталогов на пути к файлу называется *маршрутом доступа к файлам*. Перечисляемые имена подкаталогов в маршруте доступа разделяются символом "\".

При ссылке на файл маршрут включается в спецификацию файла, при этом следует за идентификатором логического диска и отделяется от имени диска символом "\".

Например, для того, чтобы обратиться к файлу PETROV.DOC, хранящемуся в каталоге BLOCKNOT, в спецификацию диска необходимо включить следующие параметры:

- Идентификатор диска (D:\).
- Маршрут доступа (\NOTES\BLOCKNOT\).
- Имя файла (PETROV.DOC).

В результате проведения вышеуказанных действий полная спецификация файла PETROV.DOC будет выглядеть следующим образом:

```
d:\notes\blocknot\petrov.doc
```

Для файла, находящегося в подкаталоге PICTURES, полная спецификация файла будет иметь вид:

```
d:\pictures\kartinka.pcx
```

Хочется отметить, что файлы в разных каталогах могут иметь одинаковые имена. Подкаталоги также могут иметь одинаковые имена, если они подчинены разным родительским каталогам.

Вводить каждый раз полную спецификацию файла довольно неудобно, поэтому ОС позволяет объявить любой каталог "рабочим", тогда к любым файлам рабочего каталога можно обращаться не указывая маршрута. При запуске системы ОС считает рабочим корневой каталог диска. Чтобы сменить рабочий каталог, необходимо применить специальную команду DOS CHDIR (от англ. Change Directory - сменить каталог); возможно применение короткого варианта этой команды - CD. Например, чтобы объявить рабочим подкаталог BLOCKNOT, необходимо применить команду

```
D:>cd \notes\blocknot
```

Если маршрут доступа начинается с символа "\", то считается, что необходимо вести начало маршрута с корневого каталога диска. Однако вы можете указывать маршрут и относительно текущего каталога. Задание маршрута относительно корневого каталога называется *абсолютным заданием маршрута*, а относительно текущего каталога -

относительным заданием маршрута. Например, вы можете перейти в подкаталог NOTES, использовав команды

```
D:>cd notes
D:>cd blocknot
```

Чтобы вернуться вновь в каталог NOTES, вы должны будете ввести одну из следующих команд:

```
D:>cd d:\notes
D:>cd ..
```

В первом варианте этой команды вы задаете абсолютный маршрут доступа к корневому каталогу, а во втором варианте вы можете использовать специальный синтаксис команды CD, где символы ".." обозначают переход на один уровень вверх к родительскому каталогу. Если же вы попытаетесь вернуться в корневой каталог, использовав относительный маршрут доступа к файлам, введя команду

```
D:>cd notes
```

то система выдаст сообщение об ошибке

```
Invalid directory (Неверный каталог)
```

Дело в том, что при использовании относительного перехода по каталогам система осуществляет поиск указанного каталога в направлении от старших каталогов к младшим, но не наоборот. Поэтому, когда вы введете команду

```
D:>cd notes
```

то ОС попытается найти подчиненный подкаталогу BLOCKNOT подкаталог NOTES и, не найдя его, выдаст сообщение об ошибке.

Очень удобным является использование в команде CD упомянутых выше символов для перемещения по различным подкаталогам диска. Например, находясь в подкаталоге NOTES, вы можете перейти в подкаталог PICTURES, введя команду

```
D:>cd ..\pictures
```

Вы можете выполнить аналогичное действие, набрав команды

```
D:>cd d:\
D:>cd \pictures
```

Команда CD позволяет также назначить рабочие каталоги на других дисках. Например, введя команду D:>CD E:\PROJECT, вы сделаете текущим подкаталог PROJECT логического диска E.

Следует отметить, что рабочий каталог всегда явно или по умолчанию задан для каждого диска. Хотя рабочие каталоги определены для каждого из дисков, только один из рабочих каталогов может быть текущим, он отображается на экране в виде отзыва ОС (A:>, B:>, C:>).

Переназначение текущего диска не связано с назначением текущего каталога. Если вы попытаетесь одновременно сменить текущий диск и перейти в какой-то подкаталог, то операционная система выведет сообщение об ошибке. Например после ввода команды

```
D:>cd c:\nc
```

система выведет сообщение

```
Bad command (плохая команда)
```

Как отмечено ранее, к файлам текущего каталога можно обращаться, не указывая полного спецификатора файла. Однако если вы хотите обратиться к файлам другого каталога или диска, то вам необходимо будет ввести полную спецификацию файла. В случае необходимости команда CD может информировать пользователя о том, какой из каталогов на диске назначен рабочим. Так, чтобы получить справку о рабочем каталоге текущего диска (например D:) нужно ввести команду CD в виде:

```
D:>cd .
```

В ответ на экране появится сообщение

```
d:\pictures
```

указывающие, что на момент ввода команды на диске D: рабочим является каталог PICTURES.

Для того, чтобы пользователь имел возможность постоянно видеть на экране указатель маршрута рабочего каталога текущего диска, этот указатель можно ввести в состав отзыва ОС. Изменить вид системного отзыва можно при помощи команды PROMPT (подробнее смотри раздел "Команды DOS"). Наиболее удобным, по мнению автора, является следующая форма команды PROMPT:

```
prompt $P$G
```

после чего подсказка DOS примет следующий вид

```
D:\PICTURES>
```

Чтобы вернуться к прежнему виду отзыва - подсказке ОС, указывающей лишь буквенное обозначение текущего диска, необходимо просто ввести команду PROMPT без параметров.

ПРОСМОТР СОДЕРЖИМОГО ПОДКАТАЛОГОВ

Просмотреть содержимое любого подкаталога можно, воспользовавшись командой DIR. При этом на экране дисплея вы увидите список подкаталогов и файлов рабочего каталога. Если вы желаете просмотреть содержимое каталога, который отличен от рабочего, необходимо ввести команду DIR с указанием абсолютного или относительного маршрута доступа к файлам. Например, чтобы при текущем рабочем подкаталоге D:\LETTERS получить список файлов подкаталога D:\NOTES\BLOCKNOT, необходимо ввести команду

```
D:\LETTERS>dir d:\notes\blocknot
```

Необходимо помнить, что для удобной работы с командой DIR при большом количестве файлов можно использовать ключи /W и /P. Подробную информацию о команде DIR можно найти в разделе "Команды DOS".

Еще одним очень важным моментом, имеющим место при работе с древовидными каталогами, является вызов программ для выполнения.

ОРГАНИЗАЦИЯ ВЫЗОВА ПРОГРАММ ПРИ РАБОТЕ С ПОДКАТАЛОГАМИ

Если вам необходимо вызвать на выполнение какую-либо программу, находящуюся в рабочем каталоге, то необходимо просто ввести ее имя. Однако если вы хотите запустить программу, находящуюся в каком-то другом подкаталоге или на другом диске, вам потребуется указать полную спецификацию файла.

Маршрут может указываться как методом указания абсолютного маршрута D:\PICTURES>E:\FILES\PROBA.EXE, так и посредством указания относительного маршрута D:\PICTURES>D:..\SYSINFO.EXE.

Однако вызывать программы указанными выше способами довольно неудобно, и для упрощения этого процесса в DOS предусмотрена специальная команда PATH. PATH задает перечисление маршрутов поиска программ и в качестве примера может быть введена так:

```
D:>path c:\;c:\dos;d:\;d:\pictures;d:\notes;d:\letters;
```

Каждый из маршрутов, перечисленных в команде, задает подкаталог, в котором будет осуществлен поиск вызываемого на выполнение файла. Сам ввод команды не вызывает никаких видимых действий, но при этом изменяется характер поиска программы, вызванной на выполнение.

Получив запрос на вызов программы, DOS начинает поиск файла с рабочего подкаталога, затем с первого подкаталога, перечисленного в команде PATH, затем во втором подкаталоге, перечисленном в коман-

де PATH и т.д., пока не будет исчерпан весь перечень маршрутов, заданных в команде PATH.

Поскольку в процессе поиска файлов переход от подкаталога к следующему подкаталогу осуществляется в порядке перечисления подкаталогов в команде PATH, то было бы целесообразным указывать маршруты каталогов в порядке хранения наиболее важных и часто используемых программ.

Как правило, если машина используется несколькими пользователями, или часто возникает необходимость менять маршруты поиска, задаваемые в команде PATH. Эта проблема чаще всего может быть решена указанием команды PATH с различными маршрутами поиска в файле AUTOEXEC.BAT для разных вариантов работы вашего компьютера.

Кроме указанного способа, DOS предоставляет вам возможность менять маршрут поиска файлов, непосредственно выполнив команду PATH во время сеанса работы. Если вам необходимо отменить установленный список маршрутов, то необходимо ввести команду

```
D:>path ;
```

После выполнения этой команды поиск файлов будет осуществляться только в рабочем подкаталоге текущего диска или диска, указанного в спецификации файла. Команда

```
D:>path
```

информирует пользователя о текущем пути поиска файлов в системе, установленного ранее командой PATH.

СОЗДАНИЕ И УДАЛЕНИЕ ДРЕВОВИДНОЙ СТРУКТУРЫ КАТАЛОГОВ

Для создания подкаталогов в DOS предусмотрена специальная команда MKDIR (Make Directory - создать каталог, сокращенное название этой команды - MD). Допустим, вам необходимо создать на диске D подкаталог с именем SYSTEM. Для этого вы должны ввести команду

```
D:>md system
```

Как видно из примера, в этой команде отсутствует идентификатор диска, и ОС при выполнении этой команды считает, что подкаталог нужно создать на текущем диске. Если вы хотите создать подкаталог на другом диске, то в команде MD необходимо указать полную спецификацию файла.

```
D:>md c:\system
```

При создании подкаталогов вы также можете использовать и те преимущества, которые дает работа с рабочими каталогами. Если, напри-

мер, подкаталог NOTES диска D: является рабочим, то вы можете создать в нем новый подкаталог EXAMPLES, использовав команду

```
D:>md examples
```

В принципе, те же самые действия можно выполнить и создав подкаталоги, имея текущим корневой каталог диска D, но обычно сначала переходят в нижележащие подкаталоги и уж затем создают в них подкаталоги. Ниже приведены примеры использования команды MD для создания подкаталога, имея текущим корневой каталог.

```
D:>md notes\examples
```

и с использованием перехода в нижестоящий каталог

```
D:>cd notes
D:>md examples
```

После того, как требуемая вам структура подкаталогов будет создана, вы можете заполнять ее и при необходимости наращивать. Следует отметить, что не стоит особенно увлекаться созданием очень разветвленной системы каталогов. Чем больше каталогов имеется на пути к файлу, тем больше времени необходимо ОС для доступа к файлу.

Удаление каталогов, созданных по ошибке или тех, надобность в которых отпала, производится командой RMDIR (Remove Directory - удалить каталог, сокращенное название RD). Все вышесказанное по отношению к команде создания подкаталогов MD имеет отношение и к команде RD.

ОС MS-DOS накладывает определенные ограничения на использование команды RD. Например, используя эту команду, вы не сможете удалить рабочий подкаталог, а также подкаталог, который содержит в себе файлы. Если вам все-таки необходимо удалить рабочий каталог, вы должны перейти из него в какой-то другой каталог, например, родительский. Если же подкаталог содержит в себе файлы, то их необходимо сначала удалить командами DEL или ERASE. Нельзя удалить также и корневой каталог.

КОМАНДНЫЕ ФАЙЛЫ

Командные файлы в DOS - это обыкновенные текстовые файлы, состоящие из строк, представляющих команды DOS. Командные файлы имеют расширение .BAT. При запуске такого файла DOS последовательно строка за строкой выполняет содержащиеся в нем команды.

Важным достоинством командных файлов является то, что они позволяют включать в себя не только команды DOS, но и специальные команды командных файлов. О командах пакетных (командных файлов) мы подробно расскажем в разделе "Команды пакетных файлов".

Ярким примером командного файла может служить файл AUTOEXEC.BAT. Однако пользователи, используя любой текстовый редактор, могут создавать свои собственные командные файлы.

Допустим, нам часто необходимо копировать файлы с диска C: подкаталога PAS на диск E: в подкаталог ARHIV. Вы можете создать командный файл, который избавит вас от частого набора команд, позволяющих осуществить это копирование.

Создадим файл KOPIR.BAT

```
c:
cd pas
copy *.* e:\arhiv
```

При вводе команды

```
kopir.bat
```

будут последовательно выполняться строки этого файла. Сначала текущим будет объявлен диск C:, а затем система объявит рабочим каталог PAS, из которого будет осуществлено копирование всех файлов (используется групповой шаблон *.*) в каталог ARHIV диска E:.

Программа DOS, анализирующая состав командного файла, воспринимает внутренние команды командных файлов, позволяющие создать в рамках командного файла нечто вроде программы с проверкой некоторых условий, анализом определенных ошибок, условными и безусловными переходами и даже позволяет применять подпрограммы. Например, такая команда командных файлов, как IF ERRORLEVEL позволяет обнаружить ошибку, возникшую в ходе выполнения программы. Более совершенный вариант приведенного выше командного файла может выглядеть следующим образом.

```
echo Вы действительно хотите скопировать файлы ?
echo 1 - Да, 2 - Нет
if errorlevel 2 goto end
c:
cd pas
copy *.* e:\arhiv
:end
```

Командные файлы широко применяются в различных программах установки (инсталляции) прикладных пакетов. Обычно такие файлы имеют имя START.BAT или INSTALL.BAT. При выполнении таких программ машина будет периодически запрашивать у вас установку очередной дискеты для копирования файлов. Такая программа обычно копирует все файлы с дискеты на жесткий диск и настраивает устанавливаемый пакет на конкретную конфигурацию компьютера.

Командные файлы также можно использовать для вывода на экран видеомонитора различных заставок и меню. Для этого в командные файлы вставляются команды ECHO и Esc-последовательности для установки цвета символов и фона. Подробно примеры управления цветом и фоном символов приведены в описании файла ANSI.SYS в разделе "Загружаемые драйверы устройств".

ПЕРЕНАПРАВЛЕНИЕ ВВОДА-ВЫВОДА

В большинстве случаев данные вводятся с клавиатуры, а выводятся на дисплей. В DOS эти два физических устройства обозначаются одним логическим устройством, называемым консолью CON:. Консоль является устройством ввода-вывода информации почти для всех команд. Однако DOS разрешает пользователю перенаправлять потоки вводимых и выводимых данных.

Предположим, что вам необходимо принять данные из последовательного порта в файл. Может потребоваться выдать результаты не на экран, а на принтер. Эти операции реализуются с помощью операций перенаправления в команде. Перенаправление потоков ввода-вывода применяется и для организации конвейеров, в которых данные с выхода одной команды перенаправляются на вход другой. Пусть вам необходимо просмотреть выход команды TYPE в виде экранных страниц; одно из решений заключается в организации конвейера, в котором выход команды TYPE подается на вход команды MORE. После ввода команды

```
C:>type c:\system\system.txt | more
```

DOS выведет на монитора содержимое файла SYSTEM.TXT из подкаталога SYSTEM диска C: и потом направит данные на вход команды MORE. Символом конвейеризации служит вертикальная черта (|).

Когда в команде вы не указываете устройство вывода, по умолчанию принимается экран, который часто называется *стандартным выводом*. Если вы хотите сохранить выход команды в файле вместо показа его на экране, следует указать символ операции перенаправления вывода - знак ">".

Например, приведенная ниже команда запоминает результаты команды DIR в файле с именем FILEOUT:

```
C:>dir > fileout
```

Поскольку перенаправление вывода в существующий файл перезаписывает его содержимое, иногда возникает необходимость добавить вывод в конец уже созданного ранее файла. Для этого применяются специальные символы ">>".

Например, команда

```
D:>dir >> fileout
```

помещает выход команды DIR в конец файла FILEOUT, находящегося в рабочем каталоге. Как и прежде, если файл FILEOUT не существует, DOS сначала создаст этот файл.

Часто возникает необходимость осуществить ввод информации не с клавиатуры, а с другого устройства. Символом перенаправления ввода служит знак "<". Он заставит DOS осуществить ввод в команду данных не с клавиатуры, а с указанного файла или устройства. Предположим, что в файле FILEOUT содержатся листинги нескольких каталогов, а вам нужно рассортировать имена файлов по алфавиту и поместить результат в другой файл FILELIST. Эту задачу решит команда

```
D:><fileout>filelist
```

По этой команде DOS считает содержимое файла FILEOUT в память, отсортирует его и передаст результат в файл FILELIST. В данном примере предполагается, что файлы FILELIST и FILEOUT находятся в рабочем каталоге.

ФИЛЬТРЫ И КОНВЕЙЕР

Фильтром называется команда, которая осуществляет загрузку данных из указанного файла, преобразует их и затем выводит их на указанное устройство вывода.

В операционной системе MS-DOS имеются 3 команды, считающиеся фильтрами. Название и краткое описание функций этих команд приведено в таблице 4.5. Подробную информацию о фильтрах вы сумеете найти в разделе "Команды DOS".

Таблица 4.5

Команды фильтров	
Название	Описание
FIND	**Отыскивает указанный текст в файле**
MORE	**Выводит содержимое файла страницами**
SORT	**Сортирует содержимое файла в алфавитном порядке**

Выходные данные после обработки фильтром разрешается перенаправлять в файл или в другую команду, используя *конвейер*. Предположим, что вам необходимо посмотреть файлы текущего каталога, отсортиро-

ванные по алфавиту; для этого символ конвейеризации должен быть указан между командой DIR и командой SORT:

```
D:>dir | sort
```

В данном случае символ конвейеризации посылает выходные данные команды DIR на вход команды SORT, и только после выполнения обеих команд данные будут выведены на экран. Этот пример не только показывает типичный пример использования конвейера, но и применения команды SORT.

В конвейерах DOS не запрещает и использование символов перенаправления. В качестве примера приведем ранее пример использования конвейерной обработки данных для команд DIR и SORT. Следующая команда поместит отсортированный листинг рабочего каталога в файл FILEOUT:

```
D:>dir | sort > fileout
```

Используя команды фильтрации и операции, вы совершенно свободно можете использовать полные спецификации файлов. Например, приведенную выше команду можно модифицировать:

```
D:>dir c:\dos | sort > fileout
```

Вообще говоря, использование операций фильтрации и перенаправления предусматривает работу со всеми возможными вариантами команд. Кроме достижения уже показанной гибкости, в одной командной строке в конвейер можно объединить и более двух команд для сложной обработки данных.

Так, при выполнении команды

```
D:>dir | sort | more
```

производится сортировка рабочего каталога и вывод результата страницами.

ВНУТРЕННИЕ КОМАНДЫ ПАКЕТНЫХ ФАЙЛОВ

Создание пакетных файлов. Как уже было отмечено ранее, командный файл можно создать при помощи любого текстового процессора, но чаще всего используют 2 основных метода создания командных файлов:

- С применением простейшего редактора, например EDIT из состава DOS.
- Вводом команды DOS “COPY CON имя_файла.bat”.

Если вы хотите организовать пакетный файл с помощью команды COPY CON, вам необходимо будет выполнить следующие действия:

- Наберите на клавиатуре команду

  ```
  copy con имя_файла.bat
  ```

  и затем нажмите клавишу Enter.
- Наберите на клавиатуре содержимое файла. По окончании ввода каждой набранной строки нажимайте клавишу Enter.
- Закончив ввод последней строки, нажмите комбинацию клавиш Ctrl-Z, обозначающую конец файла, или просто нажмите клавишу F6. Для записи файла на диск нажмите Enter. После записи файла вы увидите следующее сообщение

  ```
  1 file(s) copied
  ```

  Это сообщение говорит пользователю о том, что DOS скопировала 1 файл. Для вызова командного файла необходимо набрать его имя. Если он находится в текущем каталоге или прописан в маршрутах поиска команды PATH, то вы можете не указывать полную спецификацию файла. Расширение .BAT при вызове командного файла на выполнение указывать не обязательно.

5 Команды пакетных файлов

Условные обозначения:

диск: - имя диска, содержащего файл с командой;

\подкат - описание маршрута доступа к файлу;

имя_файла - название файла, содержащего команду;

параметры - параметры командного файла;

(!) - команда впервые введена в версию 6.0.

@

Подавление вывода изображения строки командного файла, перед которой установлен символ "@".

Формат команды:

@строка командного файла

Пример использования:

@echo off

будет выведена как сама строка ECHO OFF, так и последующие за ней в результате работы самой команды ECHO.

CALL

Команда вызывает один командный файл из другого с последующим возвратом к вызывающему командному файлу.

Формат команды:

call *диск:\подкат\имя_файла параметры*

Пример использования:

call system

Осуществляется вызов вложенного командного файла SYSTEM.BAT.

Примечания:

- Если в имени файла вы не указываете его полную спецификацию, то считается, что файл находится в текущем подкаталоге.
- Вложенный командный файл будет выполнен так же, как если бы вы ввели команду выполнения этого файла с клавиатуры.
- Командный файл имеет возможность вызывать самого себя, но тогда в нем должна быть предусмотрена возможность выхода из этого состояния. Во вложенном командном файле не допускается использование символов перенаправления потоков ввода-вывода.

(!) CHOICE

Команда подсказки и выбора варианта выполнения командного файла.

Формат команды:

choice /C:*key_symbols* /N /S /T:*c,nn текст*

Параметры:

В поле текста пользователь должен набрать тот текст, который будет высвечен при выполнение команды CHOICE.

key-symbols:

В этом поле необходимо задать ключевые символы, которые будут обрабатываться командой CHOICE.

Ключи

/C	этот ключ разрешает ввод-вывод информации, указанной в поле *key-symbols*.
/N	запрещение вывода информации из поля *key-symbols*.
/S	использование этого ключа делает команду CHOICE к верхнему или нижнему регистру. Команда CHOICE не будет выполняться, к примеру, если вы указали ключевым символом N, а пытаетесь ввести символ n.
/T:*c,nn*	изменение интервала времени, в течение которого команда CHOICE будет ожидать ввода управляющего символа. Желательно применять ключ /T вместе с ключом /C. Параметры ключа /T команды CHOICE могут принимать следующие значения:

c - установка варианта работы команды CHOICE, принимаемого по умолчанию;

nn - установка времени паузы перед тем, как можно будет ввести управляющий символ. Допустимые значения этой величины могут находиться в интервале от 0 до 99.

Пример использования:

choice /C:ync Yes, No, Continue

В результате выполнения вышеприведенной команды пользователь увидит следующую строку:

Yes, No, Continue [Y,N,C]?

Команда

choice /N Yes, No, Continue,

приведет в выводу следующей строки

Yes, No, Continue ?

Ввод команды

choice /C:ync/T:n,5

приведет к следующим результатам - будет выведена строка

[Y,N,C]?

и DOS в течение *5* секунд будет ожидать ввода управляющего символа. Если в течение указанного времени не произойдет ввода управляющего символа, то система автоматически начнет обрабатывать командный файл по варианту "N" (No).

Примечания:

Команда CHOICE обладает возможностью работы с условием ERRORLEVEL. Условие ERRORLEVEL - указывает DOS на необходимость анализа кода завершения программы. Если код завершения больше или равен указанному, то условие выполняется.

```
@echo off
cls
echo .
echo a Norton Disk Doctor
echo b Microsoft Anti-Virus
echo c Speed Disk
echo .
choice /C:abc Choose an option
if errorlevel 1 goto ndd
```

```
if errorlevel 2 goto msav
if errorlevel 3 goto spd
:ndd
ndd
goto end
:msav
msav
goto end
:spd
speedisk
goto end
:end
```

В процессе выполнения этой программы на экран будет выведен список вариантов обработки этого командного файла, и система будет ожидать ввода управляющего символа. Если управляющий символ будет введен, то с помощью команд IF ERRORLEVEL код завершения GOTO будет задан необходимый вариант работы.

Если управляющий символ не введен, то DOS будет ожидать его ввода, а если он введен неправильно, то ОС подаст звуковой сигнал и выведет символ с кодом 07h на экран.

ECHO

Выводит сообщения, а также разрешает или запрещает вывод содержимого командных файлов.

Форматы команды:

echo *сообщение* - вывод строки на экран видеомонитора.
echo off - запрет вывода строк из командного файла.
echo on - разрешение вывода строк из командного файла.
echo - вывод текущего состояния команды ECHO.

Пример использования:

```
echo off
echo Этот командный файл
echo форматирует и контролирует
echo новые диски
```

В результате выполнения этого командного файла ОС осуществит вывод на экран монитора следующего сообщения:

```
Этот командный файл
форматирует и контролирует
новые диски
```

Примечания:

- Команда ECHO OFF отменяет вывод командных строк до момента, пока не встретится команда ECHO ON или файл закончится.
- Команда ECHO влияет на вывод строк только самих внутренних команд пакетной обработки и не влияет на вывод информации команд DOS. При загрузке устанавливается по умолчанию режим ECHO ON.

FOR

Циклическое выполнение команд для набора элементов.

Формат команды:

for %% in *переменная (набор элементов)* do *команда*

Параметры:

переменная - односимвольная переменная, последовательно принимающая значения слов или имен файлов, входящих в поле *набор*.

набор элементов - одно или более символьных слов или спецификаций файлов.

команда - последовательно выполняемая для элементов набора команда DOS.

Пример использования:

Допустим, вам необходимо часто проводить копирование файлов PROG.ASM, PROG.OBJ, PROG.EXE из рабочего каталога на электронный диск E:. С целью упрощения этого процесса необходимо в соответствующем файле дать команду:

```
for %%a in (asm obj exe) do copy prog.%%a d:
```

В этом примере формальный параметр %%A последовательно сопоставляется со списком фактических параметров в круглых скобках и используется в команде COPY, которая в данном случае сработает 3 раза подряд. Подставив в качестве второго аргумента команды COPY вместо имени диска D: еще один формальный параметр %1, можно будет менять назначение копирования, задавая его с клавиатуры.

Примечание:

Если вы используете команду for...in...do, задавая ее в командной строке DOS, ее формат меняется.

```
for % in переменная (набор элементов) do команда
```

GOTO

Позволяет передавать управление на метку и тем самым осуществлять циклическое выполнение командного файла (условный переход) или создавать разветвленные участки командного файла. В последнем случае команда GOTO обычно используется с командой IF.

Формат команды:

goto *метка*

Примеры использования:

Пусть командный файл имеет вид

```
echo off
format a:/s
if errorlevel 0 goto end
echo Ошибка форматирования диска !
:end
echo Конец командного файла
```

В результате обработки этого файла система сначала осуществит форматирование дискеты в накопителе A с записью ОС. После завершения форматирования в командном файле производится анализ кода завершения программы FORMAT. Если форматирование прошло успешно, то в пакетном файле будет осуществлен переход на метку END. Если же в процессе форматирования произошли ошибки, то будет выведено сообщение "Ошибка форматирования диска !", предупреждающее пользователя о том, что процесс форматирования прошел с ошибками.

Примечания:

Для метки используются один или несколько символов. Метка должна предваряться двоеточием и начинаться с первого символа соответствующей строки. Если в процессе выполнения файла встречается метка, которая не существует, то выполнение файла прекращается и выводится аварийное сообщение. Длина метки может составлять от одного до восьми символов.

IF

Выполнение команд при наличии определенного условия

Форматы команды:

if *условие команда*
if not *условие команда*

Условия:

errorlevel - проверка кода завершения программы, сработавшей перед оператором IF. Если код завершения больше или равен указанному в ERRORLEVEL, то команда выполняется.

exist - проверка наличия файла в указанном подкаталоге.

%n - значение формального параметра (сопоставляется с конкретной строкой).

команда: в поле *команда* можно записать любую внутреннюю команду командного файла.

Примеры использования:

```
echo "Введите ваш код ?"
choice /c:1234
if errorlevel 1 goto user1
if errorlevel 2 goto user2
if errorlevel 3 goto user3
if errorlevel 4 goto user4
cd rab
:user1
cd user1
goto end
:user2
cd user2
goto end
:user3
cd user3
goto end
:user4
cd user4
goto end
:end
echo Вы находитесь в рабочем каталоге
```

В процессе выполнения этого командного файла система попросит вас ввести один из четырех возможных кодов и далее в зависимости от выбранного варианта работы осуществит переход на соответствующие метки. Далее будут выполнены команды, предписанные каждому варианту обработки командного файла.

Допустим, вам надо узнать в начале работы, не имеется ли у вас на компьютере сообщений, полученных, например, с помощью модема и хранящихся в файле MAIL.DAT. Для решения этой задачи удобно создать строку командного файла, которая будет осуществлять эту проверку.

```
if exist mail.dat copy mail.dat con
```

В процессе выполнения этой строки при обнаружении файла MAIL.DAT в текущем подкаталоге будет осуществлен его вывод на экран командой COPY MAIL.DAT CON.

Существует еще один вариант использования команды IF. Команда IF может применяться для работы с формальными параметрами. Например, необходимо вызывать текстовый редактор Лексикон и передавать ему на вход имя файла документов с расширением .TXT.

```
if %1 .==. goto nofile
e:\lex\lex %1
goto end
:nofile
echo Вы забыли ввести имя загружаемого файла
:end
```

Назовем этот файл именем RECORD.BAT и вызовем на выполнение командой RECORD TEXT.TXT. В результате работы командного файла будет выполнена следующая команда DOS:

```
e:\lex\lex text.txt
```

Примечания:

- Если при использовании условия ERRORLEVEL указанная команда не выполняется, то осуществляется переход на следующую строку командного файла.
- При использовании символьных строк следует помнить, что команда различает символы верхнего и нижнего регистров.
- Если вы используете формальных параметров %0...%9, то использование пустых строк не допускается.
- Условие NOT влияет на процесс выполнения команды. Если условие NOT присутствует, то при выполнении других условий команда не выполняется, если NOT отсутствует, то команда при выполнении других условий выполняется.

PAUSE

Команда приостановления выполнения командного файла.

Формат команды:

pause *сообщение*

сообщение - любое специальное сообщение.

Пример использования:

Предположим, вам необходимо, чтобы в процессе выполнения вашего командного файла программа выводила сообщение о смене диска в одном из накопителей. Для этого можно воспользоваться следующей командой

```
pause Вставьте в накопитель A: новую дискету
```

Примечания:

- Сообщения выводятся командой PAUSE только в режиме ECHO ON. На следующей строке после команды PAUSE и ее сообщения (если оно имеется) DOS выведет сообщение "Press any key to continue".
- Параметр *сообщение* может иметь длину до 123 символов. Вы также можете прервать выполнение командного файла в любой момент, воспользовавшись комбинацией клавиш Ctrl-C или Ctrl-Break.
- В случае, если вы прервали выполнение командного файла посредством нажатия Ctrl-Break, то ОС выведет сообщение следующего характера "Terminate you job [Y/N]?" - "Вы хотите прервать свою работу ?". Если вы нажмете клавишу "Y", то система прервет выполнение командного файла, а если вы введете "N", то ОС продолжит выполнение файла пакетной обработки.

REM

Команда позволяет задавать комментарии при выполнении командного файла.

Формат команды:

rem *сообщение*

сообщение - любая символьная строка длиной до 123 символов.

Пример использования:

```
rem Строка комментария
```

Примечание:

Следует отметить, что команда REM выводит сообщения только в режиме ECHO ON, а команда ECHO выводит сообщения всегда, независимо от режима команды ECHO.

SHIFT

Команда вызывает сдвиг списка формальных параметров влево относительно списка фактических параметров.

Формат команды:

shift

Примеры использования:

Пакетный файл COPYSET.BAT показывает использование команды SHIFT:

```
rem Файл COPYSET копирует любое количество
rem файлов в каталог.
rem Команда имеет вид:
rem copyset dir файлы
:one
if "%1"= "" goto two
set todir=%1
shift
copy %1 % todir%
goto one
:two
set todir=
echo Копирование окончено
```

Этот файл копирует список файлов в конкретный каталог.

6 Директивы файла CONFIG.SYS

BREAK

Управление реакцией DOS на нажатие клавиш Ctrl-C или Ctrl-Break.

Формат директивы:

break on
break off

Примечание:

Управлять реакцией ОС на нажатие клавиш Ctrl-C или Ctrl-Break можно также, задавая команду BREAK из командной строки DOS.

BUFFERS

Задание числа буферов DOS

Формат директивы:

buffers = *i,j/x*

Параметры:

i - указание числа буферов DOS для ускорения операций ввода-вывода данных. Смысл организации буферов ввода вывода заключается в следующем. При организации работы с файлами DOS может многократно обращаться к одним и тем же частям файла. При наличии буферов ввода-вывода DOS считывает несколько частей файла в свои буферы, организованные в оперативной памяти. При дальнейшей работе с частями файла в буферах DOS будет обращаться к ним не на диск, а в оперативную память, что, конечно, намного быстрее. Значение *i* может принимать следующие значения:

- при указании ключа *x i* меняется от 1 до 9999;
- если ключ *x* не указан, *i* меняется от 1 до 99.

Размер одного буфера ввода-вывода составляет 512 байт, то есть соответствует емкости одного сектора на диске;

j - объем буфера предввода. Буфер предввода также используется для ускорения дисковых операций. Значение *j* меняется от 1 до 8. Буфер предввода предназначен для считывания секторов на диске, непосредственно следующих за затребованным сектором. Буферы предввода особенно эффективны в случаях работы с очень длинными файлами последовательного доступа.

Ключ

/x	разрешает размещение буферов ввода-вывода и буферов предввода в расширенной памяти (EMS - Expanded memory). Однако если ваша машина имеет дополнительную память (XMS) и она поддерживается DOS с помощью драйвера HIMEM.SYS, то буферы DOS будут размещены в этой памяти. Директива CONFIG.SYS DOS имеет значение HIGH.

Примечание:

Помните, что к вопросу определения числа буферов DOS нужно подходить очень внимательно. При увеличении числа буферов DOS производительность компьютера сначала возрастет, а затем начнется снижение производительности. Это связано с тем, что в поисках нужного фрагмента файла DOS придется просматривать все большее количество файлов. Кроме того, увеличение числа буферов приводит к уменьшению объема памяти, доступной прикладным программам. Для машин с объемом оперативной памяти от 512 до 640 K рекомендуется иметь 30-40 буферов DOS. При увеличенном объеме памяти число буферов может колебаться в диапазоне от 40 до 60. Принимаемое по умолчанию число буферов находится в диапазоне от 2 до 15 и зависит от типа используемого диска.

COUNTRY

Настройка национальных форматов даты/времени и специальных символов (денежных единиц, десятичных разделителей и разделителей данных в списках).

Формат директивы:

country = *код_страны кодовая_страница*
диск:\маршрут\имя_инф_файла

Пример использования:

country = 001 437 c:\dos\country.sys

В этом примере производится настройка ОС на форматы времени/даты и специальных символов, принятые в США. Код страны 001, кодовая страница 437 - США.

Поля *диск:, маршрут, имя_инф_файла* содержат данные о местонахождении файла Country.sys, содержащего информацию о национальных форматах. Подробно информация о кодах стран и кодовых страницах приводится в описании команды KEYB в разделе "Команды DOS".

DEVICE

Подключение загружаемых драйверов устройств.

Формат директивы:

device = *диск:\маршрут\имя_файла_драйвера параметры*

Параметры:

диск:, маршрут, имя_файла_драйвера определяет полную спецификацию файла, являющегося загружаемым драйвером устройства.

параметры - это поле определяет параметры с которыми необходимо загружать драйвер устройства. Конкретно значения параметров для каждого из драйверов устройств определяются в разделе "Загружаемые драйверы устройств" или, если вы используете какое-то не стандартное устройство ввода-вывода, - в технической документации к нему.

DEVICEHIGH

Подключение драйверов устройств с их загрузкой в блоки старшей памяти (UMB).

Форматы директивы:

devicehigh = *диск:\маршрут\имя_файла_драйвера параметры*

Для загрузки драйверов в определенную часть дополнительной памяти используется следующая директива:

devicehigh = /L:область1,мин_размер1; область2,мин_размер2 /S диск:\маршрут\имя_файла_драйвера параметры

Параметры:

диск:, маршрут, имя_файла_драйвера определяет полную спецификацию файла, являющегося загружаемым драйвером устройства.

параметры - это поле определяет параметры с которыми необходимо загружать драйвер устройства. Конкретно значения параметров для каждого из драйверов устройств определяются в разделе "Загружаемые

драйверы устройств" или если вы используете какое-то нестандартное устройство ввода-вывода без технической документации к нему.

область1, область2 - здесь указываются блоки старшей памяти, которые будут использоваться драйвером после его загрузки.

мин_размер1, мин_размер2 - минимальный размер драйвера, занимаемый им в указанном блоке старшей памяти.

Ключи

/L	загрузка драйвера директивой DEVICEHIGH в свободные блоки старшей памяти;
/S	минимизация размера памяти, занимаемой драйвером.

Примеры использования:

Например, необходимо загрузить ваш драйвер MYDRIV.SYS в верхнюю область памяти на компьютере 80386. Используйте директиву:

```
devicehigh = c:\util\mydriv.sys
```

В следующем примере производится загрузка драйвера манипулятора типа "мышь" MOUSE.SYS в область верхней памяти в блок 2:

```
devicehigh = /L:2 c:\drivers\mouse.sys
```

Если необходимо загрузить драйвер MYDRIV.SYS в блок 1 старшей памяти, и вы хотите, чтобы он использовал также и 3 блок UMB, используйте директиву:

```
devicehigh = /L:1;3 c:\util\mydriv.sys
```

Для того, чтобы этот же драйвер использовал по 30 байт в каждом блоке старшей памяти (1 и 3) необходимо включить в CONFIG.SYS следующую директиву:

```
devicehigh = /L:1,30;3,30 c:\util\mydriv.sys
```

Примечания:

- Перед использованием директивы DEVICEHIGH необходимо использовать директивы
  device = c:\dos\himem.sys и dos = umb или dos = high,umb.
- Если ключ /L указан, то система загружает драйвер в полном объеме в дополнительную память в указанный блок UMB, что приводит к неэффективному использованию оперативной памяти. Если же указать минимальный размер необходимой резидентной части программы в занимаемом UMB, то можно увеличить часть памяти, доступной другим программам.

- Вы также можете возложить задачу минимизации объема памяти, занимаемой драйвером, на операционную систему, указав ключ /S. Этот же ключ использует программа оптимизации использования памяти MemMaker, когда она оптимизирует ваш файл CONFIG.SYS с целью установления системной конфигурации для наиболее оптимального использования оперативной памяти вашего компьютера. Если же вы не знаете, как правильно указать параметры *область1,мин_размер1,область2,мин_размер2* в директиве DEVICEHIGH /L:, то можно загрузить драйвер обычной директивой DEVICE, посмотреть размер установленного драйвера, задав команду MEM /M. После этого загрузите драйвер директивой DEVICEHIGH /L:.

DOS

Загрузка операционной системы в область дополнительной (XMS) памяти или в блоки старшей памяти.

Формат директивы:

dos = *параметр*

Параметры:

high - DOS будет загружена в дополнительную память (свыше 1 MB).

umb - DOS будет управлять блоками старшей памяти (UMB), которые будут доступны, как самой DOS, так и прикладным программам.

high, umb - DOS будет загружена в область дополнительной памяти, и свободные блоки старшей памяти будут доступны прикладным программам.

Примечания:

Директива DOS файла CONFIG.SYS выполняется только в том случае, если перед ней имеется директива device = himem.sys. В противном случае при загрузке системы на экране монитора появляется следующее сообщение:

```
HMA is not avaliable
DOS loaded is low
```

Некоторые программы не могут вообще работать, если не обеспечена поддержка старшей памяти.

DRIVPARM

Изменение параметров дисковых накопителей.

Формат директивы:

drivparm = /D:*номер устройства* /C /F:*тип* /H:*головки* /S:*секторы* /T:*дорожки* /I /N

Ключи

/D:	номер накопителя. Значение D принимает значения в диапазоне от 0 до 255 (0=A, 1=B, 2=C и т.д.);
/C	указывает, что дисковод имеет аппаратный контроль фиксации открытой дверцы. Это необходимо для обнаружения факта смены дискеты;
/F:*тип*	определение типа накопителя: 0 - 5.25" 160 K / 180 K или 320K / 360K; 1 - 5.25" 1.2 MB; 2 - 3.5" 720 K; 5 - винчестер; 6 - накопитель на магнитной ленте; 7 - 3.5" 1.44 MB; 8 - оптический дисковод (CD); 9 - 3.5 " 2.88 MB;
/H:*головки*	количество головок физического накопителя. Значение H меняется от 1 до 99;
/S:*секторы*	определяет число секторов на дорожке. Число секторов меняется от 0 до 99;
/T:*дорожки*	определяет число дорожек на носителе информации. Число дорожек меняется в пределах от 0 до 99;
/N	определяет блоковое устройство с несменным носителем;
/I	указывает на подключение к контроллеру гибких дисков 5.25" дисковода на 3.5". Этот ключ необходим для машин, BIOS которых не поддерживает геометрию дисководов 3.5".

Пример использования:

drivparm = /D:1 /C /F:8 /H:2 /S:34 /T:1024

Вместо накопителя на гибких дисках B: назначается оптический дисковод с головок записи-считывания 2; с 1024 дорожками и 34 секторами на каждой дорожке.

Примечание:

Указанные в директиве DRIVPARM значения должен поддерживать драйвер переопределенного устройства.

FCBS

Определение числа одновременно открытых управляющих блоков файлов.

Формат директивы:

fcbs = *x*, *y*

Параметры:

x - этот параметр показывает число одновременно открытых файлов. Величина x находится в диапазоне от 0 до 255.

y - параметр определяет число файлов, которые не могут быть автоматически закрыты системой при необходимости открытия новых файлов. Число *y* колеблется в диапазоне от 0 до 255, но не может быть больше параметра *x*.

Примечание:

Директива FCBS обычно используется в случае работы с программным обеспечением, разработанным для ранних версий DOS.

FILES

Задание числа одновременно открытых файлов.

Формат директивы:

files = *число_файлов*

Параметр *число_файлов* задает количество одновременно открытых файлов и может принимать значения от 8 до 255.

Примечания:

- По умолчанию число одновременно открытых файлов равно 8. Увеличение числа одновременно открытых файлов ускоряет работу с программами, активно работающими с большим числом файлов. Однако увеличение количества одновременно открытых файлов приводит к уменьшению объема свободной оперативной памяти.
- Для машин с объемом оперативной памяти 512 - 640 K рекомендуется иметь от 20 до 30 одновременно открытых файлов. В машинах с оперативной памятью более 640 K число одновременно открытых может быть увеличено до 40.

INSTALL

Установка внешних команд (утилит) DOS в файле CONFIG.SYS.

Формат директивы:

install *диск:\маршрут\имя_утилиты_DOS команда*

Параметры:

диск:\маршрут\имя_утилиты_DOS - полный идентификатор файла, содержащего в себе внешнюю команду DOS.

команда - параметры, которые необходимо передавать внешней команде DOS при ее запуске.

Примечания:

В качестве внешних команд DOS возможно, например, использование следующих утилит: Fastopen.exe, Keyb.com, Nlsfunc.exe, Share.exe.

LASTDRIVE

Определяет максимальное количество логических дисков, которое может иметь компьютер.

Формат директивы:

lastdrive = *символ*

Параметры:

символ - буква латинского алфавита от A до Z, задающая букву последнего логического диска, который может иметь компьютер.

Пример использования:

lastdrive = h

Вышеприведенная директива указывает, что компьютер может иметь 8 логических дисков (от A до H).

REM

Включение комментария в строки CONFIG.SYS

Формат директивы:

rem *сообщение*

Параметры:

сообщение - любая символьная строка.

Примечания:

Директиву REM удобно использовать для временного исключения из файла CONFIG.SYS без их физического исключения.

SHELL

Указание места расположения в системе командного процессора.

Формат директивы:

shell = *диск:\маршрут\имя_файла диск2:\маршрут2\имя_файла2 /ключи*

Параметры:

диск:\маршрут\имя_файла - полный идентификатор файла, содержащий в себе месторасположение командного процессора.

диск2:\маршрут2\имя_файла2 - полный идентификатор файла, содержащий в себе месторасположение 2 командного процессора.

Ключи

/E:	устанавливает размер системного окружения. Размер системного окружения может принимать значение в пределах от 160 до 32768. Значение должно быть кратно 16;
/P	эта директива указывает, что командный процессор полностью резидентен в памяти, и из него нельзя выйти командой EXIT;
/MSG	эта директива предписывает командному процессору сохранять в оперативной памяти тексты аварийных сообщений. Ключ /MSG используется совместно с ключом /P;
/K	предписывает выполнять специальные программы или командные файлы, подобные AUTOEXEC.BAT при запуске приложений в Windows. Для понимания этого вопроса необходимо посмотреть Pif Editor (файл DOSPRMT.PIF) и задать ключ /K в Optional Parameters Box. Однако не рекомендуется применять эту директиву во избежание проблем, которые могут возникнуть при запуске DOS-приложений в среде Windows;
/C	указанные параметры для первого командного процессора передаются второму командному процессору.

Пример использования:

```
shell = c:\dos\command.com /E:1024 /P
```

В результате выполнения этой директивы после загрузки системы в системе активизируется командный процессор, находящийся на диске C в подкаталоге DOS. Размер системного окружения увеличивается до 1 K. Кроме этого объявляется, что командный процессор резидентен в памяти. Так как спецификация второго командного процессора опущена, то в файле AUTOEXEC.BAT следует указать следующую строку

```
set comspec = c:\dos\command.com
```

которая позволит подгружать в память нерезидентную часть командного процессора в случае затирания ее прикладными программами.

Примечания:

Размер системного окружения для командного процессора устанавливается равным 256 байт.

STACKS

Задание числа стеков DOS.

Формат директивы:

stacks = *число_стеков,размер_стека*

Параметры:

число_стеков - количество стеков, используемых DOS, в процессе обработки прерываний. Число стеков может варьироваться в пределах от 8 до 64.

размер_стека - длина одного стека в байтах. Длина стека может изменяться от 32 до 512 байт.

Пример использования:

```
stacks = 9,256
```

Вышеприведенной директивой объявляется, что ОС должна использовать для обработки прерываний 9 стеков, размером по 256 байт каждый.

Примечания:

Директива обычно применяется для программ, активно использующих внешние прерывания, например Windows и используется преимущественно на компьютерах, оснащенных процессором Intel 286 и выше.

SWITCHES

Задание способа взаимодействия DOS с расширенной клавиатурой машин IBM PC/AT.

Формат директивы:

switches = /W /K /N /F

Ключи

/W	при работе DOS будет использовать файл WINA20.386 для организации взаимодействия с Windows 3.0. Файл WINA20.386 должен находится в корневом каталоге системного диска;
/K	запрещается поступление с расширенной 101-клавишной клавиатуры дополнительных кодов (F11, F12 и др.). Другими словами, при использовании этого ключа расширенная клавиатура IBM PC/AT функционирует, как стандартная клавиатура машин IBM PC;
/N	запрещается использование клавиш F5 и F8 для пропуска выполнения директив CONFIG.SYS и команд AUTOEXEC.BAT при начальной загрузке ОС;
/F	организация пропуска двухсекундной задержки перед появлением сообщения "Starting MS-DOS ..." при загрузке операционной системы.

(!) СПЕЦИАЛЬНЫЕ ДИРЕКТИВЫ ЗАДАНИЯ РАЗЛИЧНЫХ СПОСОБОВ НАСТРОЙКИ СИСТЕМЫ В CONFIG.SYS

Одним из наиболее важных усовершенствований, появившихся в MS-DOS 6.0, является возможность настройки ОС для различных вариантов работы вашего компьютера. Мы уже затрагивали эту тему в разделе, посвященном описанию файлов CONFIG.SYS и AUTOEXEC.BAT.

Блоки директив обозначаются следующим образом:

[*текст*]

где поле *текст* содержит наименование блока директив.

Имя блока директив может быть длиной до 70 символов.

В обозначении блока директив недопустимо использование следующих символов: пробелов, \, /, :, ;, =, [,].

Стандартными обозначениями блоков директив являются [menu] и [common]. [menu] должно предварять собой директивы MENUITEM и другие описывающие варианты обработки файла CONFIG.SYS.

Директива [common] содержит в себе директивы, которые обязательно выполняются независимо от варианта работы.

Всего в MS-DOS включено 6 новых директив мультиконфигурации.

INCLUDE

Включение в блок директив одного из вариантов работы другого блока директив

Формат директивы:

include = *имя_блока*

Параметр *имя_блока* содержит имя того блока директив, который необходимо включить в какой-то другой блок команд.

Пример использования:

```
[menu]
menuitem = base_config,Base Configuration
menuitem = normal_config,Normal Configuration
menuitem = network_config,Network Configuration
[base_config]
device = c:\dos\himem.sys
dos = high
[normal_config]
include base_config
dos = umb
device = c:\dos\emm386.exe ram
device = c:\dos\ramdrive.sys 512
[network_config]
include normal_config
devicehigh = c:\dos\network.sys
[common]
break on
files = 40
buffers = 30
shell = c:\dos\command.com /E:1024 /P
```

В результате обработки этого файла система выведет стартовое меню запуска, в котором будут указаны три варианта работы:

- Base Configuration.
- Normal Configuration.
- Network Configuration.

Если будет выбран вариант 1, то выполнятся директивы, упомянутые в блоке [base_config] и далее в блоке [common]. Блок [common] выполняется для всех блоков директив.

Для варианта 2 система, благодаря директиве INCLUDE, автоматически выполнит директивы, значащиеся в блоке [base_config] и далее те директивы, которые находятся после директивы INCLUDE.

При выборе 3 варианта работы аналогичным образом выполнится блок директив [normal_config], который включает в себя директивы [base_config].

MENUCOLOR

Задание цвета текста и фона в стартовом меню запуска.

Формат директивы:

menucolor = *x*,*y*

Параметры:

x - задает код цвет символов из приводимого далее списка возможных кодов.

y - задает код цвета фона.

Пример использования:

```
[menu]
menucolor = 3,0
menuitem = normal
menuitem = hard_disk_cache
menudefault = normal
[normal]
device = c:\dos\himem.sys
dos = high,umb
[hard_disk_cache]
include = normal
devicehigh = c:\app\mydriv.sys
[common]
break on
files = 40
buffers = 30
shell = c:\dos\command.com /E:1024 /P
```

В приведенном выше примере в блоке директив [menu] директива MENUCOLOR устанавливает следующий режим вывода текстовой информации: голубой цвет (код 3) символов на черном фоне (код 0).

Примечания:

Необходимо выбирать те цвета, которые не будут слишком сильно раздражать ваши глаза. По умолчанию принимаются следующие значения *x*=8, а *y*=0. Заданные в MENUCOLOR цветовые атрибуты будут установлены и в процессе работы вашей вычислительной системы до тех пор, пока вы не введете команду CLS, или цветовые атрибуты не

будут изменены прикладной системой. Команда CLS изменяет текущие цветовые атрибуты на принимаемые по умолчанию.

Коды цветов			
0	Black (черный)	8	Gray (серый)
1	Blue (синий)	9	Bright Blue (ярко-синий)
2	Cyan (голубой)	10	Bright Green (ярко-зеленый)
3	Green (зеленый)	11	Bright Cyan (ярко-голубой)
4	Red (красный)	12	Bright Red (ярко-красный)
5	Magenta (фиолетовый)	13	Bright Magenta (ярко-фиолетовый)
6	Brown (коричневый)	14	Yellow (желтый)
7	White (белый)	15	Bright White (ярко-белый)

MENUDEFAULT

Установка пункта стартового меню DOS по умолчанию при загрузке системы.

Формат директивы:

menudefault = *имя_блока,времен_задержка*

Параметры:

имя_блока - имя блока директив, принимаемое для выполнения по умолчанию.

времен_задержка - установка временной задержки перед выполнением блока директив по умолчанию. Параметр необязательный. Величина временной задержки меняется от 0 до 90.

Пример использования:

```
menudefault = normal,5
```

В результате выполнения этой директивы курсор, подсвечивающий пункты меню, установится на пункт NORMAL и появится справа от "Enter a choice:" следующее сообщение

```
Time remaining:05
```

Далее начнется отсчет времени ожидания ввода какого-либо варианта работы из стартового меню DOS.

Если в течение времени, заданного параметром *времен_задержка*, вы не предпримите никаких действий, то произойдет выбор варианта NORMAL и соответственно обработка CONFIG.SYS по этому варианту.

Объявленный через menuitem пункт должен быть обязательно описан ниже через [пункт] иначе будет ошибка

MENUITEM

Задание варианта обработки файла CONFIG.SYS в стартовом меню DOS.

Формат директивы:

menuitem = *имя_блока,текст*

Параметры:

имя_блока - имя блока директив, задающего вариант обработки файла CONFIG.SYS.

текст - любая символьная строка длиной до 70 символов. Данный параметр не является обязательным, в случае, если он опущен, то в качестве текста DOS будет использовать имя блока директив.

Пример использования:

```
[menu]
menuitem = base_config,Base configuration
menuitem = full_config,Normal configuration
menuitem = net_config,Normal configuration with network
```

В этом примере первым пунктом меню установлен [base_config] с выводом сообщения "Base configuration only" в стартовом меню, вторым пунктом [full_config] и третьим [net_config] с выводом соответствующих сообщений. Когда MS-DOS выведет стартовое меню, DOS будет иметь следующий вид:

```
MS-DOS 6 Startup Menu
=====================

   1. Base configuration only
   2. Normal configuration
   3. Normal configuration with network

Enter a choice: 1
```

Выбрав один из вариантов, вы зададите вариант обработки системой файла CONFIG.SYS.

SUBMENU

Организация для одного или нескольких пунктов стартового меню DOS подменю. Подменю - это меню, подчиненное главному меню. Главное меню - это меню, определенное директивой MENUITEM.

Формат директивы:

submenu = *имя-блока, текст*

Параметры:

имя-блока - наименование пункта меню, для которого необходимо организовать подменю.

текст - любая символьная строка длиной до 70 символов.

Пример использования:

```
[menu]
menuitem base_config,Base configuration
menuitem full_config,Normal configuration
submenu netmenu, Normal configuration with network

[netmenu]
menuitem lanman, Start Microsoft LAN Manager
menuitem vines, Start Banyan VINES

[base_config]
dos = high
device = c:\dos\himem

[full_config]
include = base_config
dos = umb
device = c:\dos\emm386 ram
device = c:\dos\ramdrive.sys 512

[lanman]
include = full_config
rem Commands for Microsoft LAN Manager

[vines]
include = full_config
rem Commands for Banyan VINES
```

В этом файле CONFIG.SYS организованы 3 варианта работы в стартовом меню DOS. Если выберете первых два пункта этого меню [full_config] или [base_config], то произойдет просто обработка CONFIG.SYS по этим вариантам работы.

Если же вы выбрали третий вариант работы, то вы увидите еще одно меню, которое включает в себя в свою очередь еще 2 пункта. Они называются: Start Microsoft LAN Manager и Start Banyan VINES. Выбрав один из них, вы свою очередь задействуете обработку ОС блоков директив, соответствующих выбранному режиму работы.

Внешний вид сообщений, выводимых DOS в процессе обработки приводимого файла CONFIG.SYS, приведены ниже.

```
MS-DOS 6 Startup Menu
=====================

    1. NORMAL
    2. WINDOWS
    3. XMS

Enter a choice: 1



F5=Bypass startup files   F8=Confirm each CONFIG.SYS line [N]
```

Если выбран 3 вариант работы, в главном меню появятся следующие сообщения:

```
MS-DOS 6 Startup Menu
====================

1. Start Microsoft LAN Manager
2. Start Banyan VINES

Enter a choice: 1
```

NUMLOCK

Управление состоянием индикатора NumLock.

Формат директивы:

numlock = on
numlock = off

```
[menu]
menuitem = base_config,Base configuration only
menuitem = full_config,Normal configuration
menuitem = network,Normal configuration with network
menudefault = full_config

numlock = off
```

В процессе обработки этого фрагмента файла CONFIG.SYS ОС выведет стартовое меню и погасит индикатор NumLock при загрузке системы.

7 Загружаемые драйверы устройств

ANSI.SYS

Расширенное управление клавиатурой и дисплеем. Обеспечивает дополнительные функции управления дисплеем: возможность чтения текущего положения дисплея, установка цвета символов и фона, позиционирование курсора, переназначение клавиш.

Формат директивы подключения драйвера

device = *диск:\маршрут* ansi.sys /X /K /R
devicefigh = *диск:\маршрут* ansi.sys /X /K /R

Параметры:

диск:\маршрут - указание полного пути к файлу ANSI.SYS.

Ключи

/X	переназначение отдельных клавиш (правые Alt и Ctrl, клавиши управления курсором) на дополнительной клавиатуре машин IBM PC/AT. Без этого ключа отдельные клавиши управления курсором воспринимаются, как клавиши "Home/7", "PgUp/9" и т.д.;
/K	использование алгоритма работы со 101-клавишной клавиатурой, как с 83-клавишной клавиатурой машин IBM PC. Ключ /K идентичен использованию команды SWITCHES с тем же ключом;
/R	драйвер ANSI.SYS будет производить нормализацию строк прокрутки при работе программ чтения текстов с экрана, которые могут изменять строки прокрутки.

Пример использования:

```
device = c:\dos\ansi.sys /K
```

В результате использования этой директивы к ОС будет подключен драйвер ANSI.SYS, который будет использовать интерфейс связи с клавиатурой, применяющийся в 83-клавишной клавиатуре ПЭВМ IBM PC.

Примечание:

Подробное описание команд драйвера ANSI.SYS и примеры их использования приведены в приложении 1 "Команды драйвера ANSI.SYS".

CHKSTATE.SYS

Этот драйвер используется программой оптимизации распределения оперативной памяти MemMaker, для организации процесса проверки использования памяти прикладными программами.

Формат директивы подключения:

device = *диск:\маршрут* chkstate.sys

диск:\маршрут - описание полного пути доступа к файлу CHKSTATE.SYS. Задание пути доступа к CHKSTATE.SYS выполняется автоматически при запуске MemMaker. Файл CHKSTATE.SYS должен находится в одном каталоге с MemMaker.

Примечание:

Во время оптимизации MemMaker добавляет CHKSTATE.SYS в начало вашего CONFIG.SYS и просит произвести перезагрузку вашего компьютера.

Когда процесс оптимизации закончен, MemMaker удаляет CHKSTATE.SYS из CONFIG.SYS.

Более подробную информацию об использовании утилиты MemMaker можно получить в разделе "Команды DOS".

DBLSPACE.SYS

Организация сжатия данных при работе с дисковыми накопителями.

Формат директивы подключения:

device =
devicehigh = *диск:\маршрут* dblspace.sys /MOVE

Параметры:

диск:\маршрут - описание полного пути доступа к файлу DBLSPACE.SYS.

Ключи

/MOVE перемещение модуля DOS DBLSPACE.BIN на старшие адреса доступной DOS оперативной памяти.

Пример использования:

device = c:\dos\dblspace.sys /MOVE

Организуется сжатие данных при работе с дисками,перемещение DBLSPACE.BIN с младших адресов памяти, занимаемых самой DOS и драйверами устройств на старшие адреса памяти.

Примечание:

В процессе использования DBLSPACE.SYS активизируется модуль DOS DBLSPACE.BIN, который вместе с DBLSPACE.SYS организует сжатие данных.

DISPLAY.SYS

Расширение возможностей стандартного драйвера консоли CON, входящего в состав DOS, или ANSI.SYS, если он установлен. Этот драйвер применяется для переключения кодовых страниц командой DOS MODE.

Формат директивы подключения:

device = *диск:\маршрут* display.sys con = *(тип, ном_стр, кол_стр, кол_шр)*
devicehigh = *диск:\маршрут* display.sys con = *(тип, ном_стр, кол_стр, кол_шр)*

Параметры:

диск:\маршрут - описание полного пути доступа к файлу DISPLAY.SYS.

тип - тип дисплея, установленного на компьютере. Параметр тип принимает значения EGA и LCD. EGA используется для адаптеров EGA и VGA, а LCD для жидкокристаллического дисплея.

ном_стр - номер кодовой страницы, являющейся аппаратно поддерживаемой для данного типа дисплея. Номера кодовых таблиц приведены в описании команды DOS KEYB.

кол_стр - этот параметр указывает количество страниц, которые будут использоваться при выводе текстовой информации на экран (исключая аппаратную). По умолчанию количество страниц установлено для адаптера EGA равным 6, а для адаптеров LCD равным 1.

кол_шр - количество таблиц шрифтов, аппаратно поддерживаемых для каждой кодовой таблицы. Для LCD число шрифтов равно 2, для EGA - 6.

Пример использования:

device = c:\dos\display.sys con = (ega,437,2)

В результате выполнения этой директивы объявляется использование в адаптере EGA кодовой страницы 437 (США) и устанавливается число страниц для вывода текстовой информации.

Наличие этой директивы позволяет использовать команду MODE для загрузки кодовых таблиц с национальными символами в знакогенератор дисплея.

Примечание:

В качестве параметра тип в драйвере DISPLAY.SYS возможно использование следующих аббревиатур: CGA и MONO, но это не позволит вам менять кодовые страницы дисплея, так как эти адаптеры не предусматривают переключение кодовых страниц.

DRIVER.SYS

Определение дополнительных НГМД, непосредственно не поддерживаемых DOS.

Формат директивы подключения:

device = *диск:\маршрут* driver.sys /D:*номер* /C /F:*специф* /H:*голов* /S:*секторов* /T:*дорож*

Параметры:

диск:\маршрут - описание полного пути доступа к файлу DRIVER.SYS.

Ключи

/D:*номер*	номер физического накопителя на гибких дисках (0=A, 1=B и т.д.). Номер накопителя может задаваться в диапазоне от 0 до 127. Те накопители, которые непосредственно DOS не поддерживает, должны иметь номер 2 или больший. Если ваш компьютер имеет один накопитель, то логические диски A и B имеют номер 0, в случае если компьютер имеет 2 и более накопителей диск B обозначается как 1;
/C	этот ключ указывает, что дисковод имеет аппаратную поддержку факта открытия дверцы накопителя. Ука-

занное свойство используется некоторыми программами для обнаружения факта смены дискеты;

/F:*специф* спецификатор дисковода, для указания типа используемого накопителя. Использование ключа /F: делает необязательным использование ключей /T:, /H:, /S: необязательным. Спецификаторы дисководов, используемые с ключом /F:, соответствуют типам НГМД, которые приведены ниже:

0 - 160/180 K или 320/360 K;

1 - 1.2 MB;

2 - 720 K (3.5") или (5.25");

7 - 1.44 MB;

9 - 2.88 MB;

По умолчания спецификатор дисковода равен 1;

/H:*голов* число головок накопителя. Число головок меняется в диапазоне от1 до 99. По умолчанию число головок равно 2;

/S:*секторов* число секторов на дорожке. Число секторов меняется в диапазоне от 1 до 99. По умолчанию число секторов принимаемое для типов дисководов, определенных с помощью ключа /F:, приведено ниже:

/F:0/S:9

/F:1/S:15

/F:2/S:9

/F:7/S:18

/F:9/S:36

/T:*дорож* количество дорожек на 1 стороне. Число дорожек меняется в пределах от 1 до 999. Для накопителя, определенного ключом /F:2, число дорожек по умолчанию принимается равным 80, а для определенного ключом /F:1 число дорожек принимается равным 40.

Пример использования:

```
device = c:\dos\driver.sys /D:2 /F:7
```

В процессе загрузки DOS к ней подключается дополнительный внешний дисковод C, который является накопителем с емкостью 1,44 MB (3.5");

Примечания:

Если ваш накопитель имеет нестандартную геометрию (нестандартное число головок, дорожек, секторов), вы можете использовать DRIVER.SYS, предварительно изучив техническую документацию к накопителю или обратившись к его поставщику.

EGA.SYS

Организация записи и восстановления изображения на дисплее при работе оболочки MS-DOS Task Swapper с EGA монитором.

Формат директивы подключения:

device = *диск:\маршрут* ega.sys

Параметры:

диск:\маршрут - описание полного пути доступа к файлу EGA.SYS.

Примечание:

Если вы хотите организовать в вашей системе с EGA-монитором поддержку манипулятора типа "мышь", необходимо подключить драйвер EGA.SYS перед подключением вашего драйвера "мыши".

EMM386

Поддержка расширенной памяти и эмуляция дополнительной памяти, как расширенной на машинах с процессорами 80386 и выше. EMM386 дает возможность загрузки программ и драйверов устройств в область старшей памяти (UMB).

Формат директивы подключения:

device = *диск:\маршрут* **emm386.exe** *ON/OFF/AUTO память* MIN = *размер* W = *ON/OFF* M*x* FRAME = *адрес* /P*mmmm* Pn = *адрес* X = *mmmm-nnnn* I = *mmmm-nnnn* B = *адрес* L = *minXMS* A = *альт_рег* H = *канал* D = *nnn* RAM = *mmmm-nnnn* NOEMS NOVCPI HIGHSCAN VERBOSE WIN = *mmmm-nnnn* NOHI ROM = *mmmm-nnnn* NOMOVEXBDA ALTBOOT

Параметры:

диск:\маршрут - описание полного пути доступа к файлу EMM386.EXE.

ON/OFF/AUTO - ключи, управляющие работой драйвера в целом. ON - включение поддержки расширенной памяти. OFF - выключение поддержки драйвером расширенной памяти при условии, что ее не исполь-

зуют прикладные программы. AUTO - автоматическое управление поддержкой расширенной памяти; поддержка расширенной памяти выключена до тех пор пока прикладная программа не затребует расширенную память.

память - установление объема дополнительной памяти, которая будет отображаться как расширенная память драйвером EMM386. В этом случае EMM386 использует интерфейс виртуального контроля программ (EMS/VCPI). Размер этого параметра может меняться от 64 до 32768 K блоками по 16 K. По умолчанию этот параметр принимает значение - вся доступная расширенная память. Если вы используете ключ NOEMS, этот параметр равен 0.

Ключи

MIN = *размер* — указание минимального размера доступной EMM386, с использованием (EMS/VCPI), памяти в килобайтах; EMM386 резервирует этот блок расширенной памяти после его загрузки директивой "DEVICE = EMM386" CONFIG.SYS для использования механизма (EMS/VCPI). Значения параметра *размер* изменяются в пределах от 0 до значения, указанного в параметре *память*. По умолчанию значение параметра *размер* равно 256. В случае использования ключа NOEMS это же значение равно по умолчанию 0;

W = *ON/OFF* — разрешение/запрещение использования EMM386 математического сопроцессора Weitek. По умолчанию ключ W установлен в положение OFF;

M*x* — указание адреса кадра страниц EMS. Допустимые значения x лежат в пределах от 1 до 14. Значения x, соответствующие определенным кадрам страниц EMS в шестнадцатеричном формате, приведены ниже:

1	=>	C000h	8	=>	DC00h
2	=>	C800h	9	=>	E000h
3	=>	C800h	10	=>	8000h
4	=>	CC00h	11	=>	8400h
5	=>	D000h	12	=>	8800h
6	=>	D400h	13	=>	8C00h
7	=>	D800h	14	=>	9000h

Значения, лежащие в пределах от 10 до 14, должны быть использованы для компьютеров, имеющих 512 K памяти;

FRAME = *адрес* — сегментный адрес кадра страниц EMS 3.2. Допустимые значения адреса находятся в диапазоне от 8000h до 9000h и от C000h до E000h блоками по 400h. Этот адрес не может использоваться другими программами или устройствами;

/P*mmmm* — установление адреса кадра страницы EMS 3.2. Допустимые значения этого параметра находятся в диапазоне от 8000h до 9000h и от C000h до E000h блоками по 400h;

Pn = *адрес* — установка сегментного адреса страницы EMS 4.0. Допустимые значения "n" от 0 до 255. Допустимые значения адреса находятся в диапазоне от 8000h до 9000h и от C000h до E000h блоками по 400h. Сегментные адреса страниц с 0 по 3 должны соответствовать сегментным адресам страниц версии 3.2 Lotus/Intel/Microsoft Expanded Memory Specification (LIM EMS - спецификация фирм Lotus/Intel/Microsoft на расширенную память). Если используется ключ Mx, ключ FRAME с ключом /Pmmmm вы не можете указать сегментные адреса страниц от 0 до 3 с ключом /Pmmmm;

X = *mmmm-nnnn* — установка диапазона адресов, которые не должны использоваться EMS. Значение *mmmm* устанавливает нижний диапазон адресов, а *nnnn* - верхний. Значения *mmmm* и *nnnn* должны быть указаны в шестнадцатеричной форме;

I = *mmmm-nnnn* — установка диапазона адресов, которые могут использоваться для обращения к EMS или UMB. Эти адреса не должны использоваться другими программами или устройствами. Допустимые значения *mmmm* и *nnnn* лежат в пределах от A000h до FFFFh и занимают также ближайшие 4 K памяти с меньшими адресами. Использование ключа X должно предшествовать использованию ключа I если используется два диапазона адресов;

B = *адрес* — указание нижнего сегментного адреса, доступного банкам EMS, размещенным по 16 K. Допустимые значения адресов от 1000h до 4000h. По умолчанию принимается значение адреса, равное 4000h;

L = *minXMS* — установка минимального размера дополнительной памяти XMS, доступной после загрузки EMM386;

A = *альт_рег* — установка числа быстрых альтернативных регистров, которые будут доступны EMM386. Эти регистры используются для организации многозадачной работы.

Допустимое число альтернативных регистров находится в диапазоне от 0 до 254. По умолчанию число альтернативных регистров равно 7. Каждый установленный альтернативный регистр расходует 200 байт памяти;

H = *канал* — установка числа каналов для работы EMM386. Допустимые значения каналов от 2 до 255. По умолчанию число каналов принято равным 64;

D = *nnn* — указание числа K, которые будут зарезервированы для каналов прямого доступа в память (DMA) при работе с накопителями на гибких дисках. Значение *nnn* указывается в K и меняется от 16 до 256. По умолчанию *nnn* равно 16;

RAM = *mmmm-nnnn* — указание диапазона сегментных адресов, используемых блоками старшей памяти и для поддержки также EMS. Если параметр *mmmm-nnnn* не указан EMM386 использует всю доступную старшую память для размещения UMB и создания страниц EMS;

NOEMS — драйвер EMM386 перестает управлять расширенной памятью, но продолжает поддержку блоков UMB;

NOVCPI — запрещение поддержки VCPI-приложений. Этот ключ может совместно использоваться с ключом NOEMS. Если используете ключ NOVCPI вместе без ключа NOEMS EMM386 не будет поддерживать VCPI-приложения. Если же вы используете оба ключа, EMM386 будет игнорировать значение параметра память и ключа MIN. Запрещение поддержки VCPI-приложений приводит к уменьшению объема доступной расширенной памяти;

HIGHSCAN — задание для EMM386 режима проверки доступной старшей памяти для размещения блоков старшей памяти и окон EMS. Этот ключ используется в тех случаях, когда EMM386 используется неэффективно для управления UMB и EMS;

VERBOSE — вывод драйвером EMM386 на дисплей своего текущего статуса и сообщений об ошибках в процессе загрузки EMM386 в память. По умолчанию, EMM386 выводит на экран сообщения о своих внутренних установках и создаваемых в системе ошибочных ситуациях. Возможно использование вместо VERBOSE ключа V. Для отключения вывода сообщений, нажмите и удерживайте клавишу ALT, до тех пор пока EMM386 не загрузится в память;

WIN = *mmmm-nnnn*	резервирование диапазона сегментных адресов для Windows вместо EMM386. Ключ WIN должен использоваться после указания ключей RAM, ROM и I;
NOHI	при использовании этого ключа EMM386 не будет загружен в старшую память. В нормальных условиях EMM386 всегда загружается в старшую память. Использование этого ключа уменьшает объем свободной основной памяти, но увеличивает объем доступной старшей памяти, что позволяет разместить больше блоков UMB;
ROM = *mmmm-nnnn*	при использовании этого ключа EMM386 загружает в указанный диапазон адресов теневую ROM BIOS. Значения *mmmm* и *nnnn* находятся в диапазоне адресов от A000h до FFFFh и занимают также ближайшие 4 K памяти на младших адресах. Используете этот ключ для повышения скорости работы вашей системы, если нет возможности использовать теневую RAM. При использовании этого ключа EMM386 перемещает данные из области данных BIOS в область старшей памяти;
ALTBOOT	при использовании этого ключа EMM386 использует альтернативный канал для рестарта вашей системы, когда вы нажимаете клавиши Ctrl-Alt-Del. Используйте этот ключ, когда у вас возникает необходимость перезапустить компьютер при появлении необычных сообщений и других необычных событиях, когда EMM386 загружен, или нажмите Ctrl-Alt-Del.

Примеры использования:

Для запуска эмулятора расширенной памяти необходимо добавить в CONFIG.SYS следующую строку:

```
device = c:\dos\emm386.exe
```

Допустим, вам необходимо, чтобы EMM386 было доступно 4096 K памяти и из них гарантированно 256 K (по умолчанию). Вы должны указать в CONFIG.SYS следующую строку:

```
device = c:\dos\emm386.exe 4096
```

Для эмуляции расширенной памяти, начинающейся с сегментного адреса D000h, для кадра страницы EMS и организации 512 K, доступных EMM386, используйте одну из приведенных ниже директив:

```
device = c:\dos\emm386.exe 512 frame = d000
device = c:\dos\emm386.exe 512 p0 = d000 p1 = d4000 p2 = d8000
p3 = dc00
```

Для того, чтобы старшая память не могла работать с EMS/VCPI-прикладными программами добавьте в CONFIG.SYS, приведенную ниже строку:

```
device = c:\dos\emm386.exe NOEMS NOVCPI
```

Для организации доступа ко всей старшей памяти и разрешения использования ее EMS/VCPI-прикладными программами используйте следующую строку:

```
device = c:\dos\emm386.exe  RAM
```

Примечания:

- Для использования драйвера EMM386 необходимо, чтобы драйвер HIMEM.SYS был загружен до драйвера EMM386.EXE. Для загрузки HIMEM.SYS используйте директиву DEVICE = HIMEM.SYS.
- После загрузки драйвера EMM386 возможно управление им из командной строки DOS с помощью команды EMM386. Подробности использования команды EMM386 отражены в разделе "Команды DOS".

HIMEM.SYS

Поддержка дополнительной памяти на машинах с процессором 80286 и выше.

Формат директивы подключения:

device = *диск:\маршрут*\himem.sys /A20CONTROL = ON|OFF /CPULOCK = ON|OFF /EISA /HMAIN = *m* /INT15 = *xxxx* /NUMHANDLES = *n* /MACHINE:*xxxx* /SHADOWRAM:ON|OFF /VERBOSE

Параметры:

диск:\маршрут - описание полного пути доступа к файлу HIMEM.SYS.

Ключи

/A20CONTROL = ON|OFF	значение ON заставляет HIMEM всегда держать под контролем адресную линию A20. В случае если ключ A20CONTROL установлен в положение OFF, HIMEM не будет контролировать линию A20 только тогда, когда другая прикладная программа держит A20 под контролем;
/CPULOCK = ON|OFF	ключ ON информирует драйвер о том, что он изменяет тактовую частоту компьютера. По

умолчанию ключ CPULOCK установлен в положение OFF;

/EISA — используя этот ключ, HIMEM.SYS будет управлять всей доступной старшей памятью. Особенно эффективно использование этого ключа в компьютерах с шинной архитектурой EISA (Extended Industry Standard Architecture), которые обеспечивают работу со старшей памятью свыше 16 MB; на других компьютерах HIMEM.SYS сразу получает в свое распоряжение всю доступную старшую память;

/HMAIN = *m* — устанавливает в килобайтах, области старшей памяти, который должна запросить прикладная программа, чтобы получить доступ к старшей памяти. Допустимые значения *m* лежат в пределах от 0 до 63. По умолчанию значение *m* равно 0. Это означает, что первая же прикладная программа получит доступ к старшей памяти;

/INT15 = *xxxx* — устанавливает размер памяти, резервируемой за прикладными программами, которые используют прерывание INT 15. Значения *xxxx* меняются от 64 до 65535. Значения *xxxx*, меньшие 64 K, рассматриваются как равные 0;

/NUMHANDLES = *y* — устанавливает максимальное число дескрипторов блоков дополнительной памяти (EMB). Допустимые значения *y* лежат в пределах от 1 до 128; По умолчанию значение *y* = 32. Каждый добавленный дескриптор блоков расходует 6 байт оперативной памяти. Использование ключа NUMHANDLES неэффективно при работе Windows в расширенном режиме работы процессора 80386;

/MACHINE:*xxxx* — объявление типа компьютера. Обычно HIMEM.SYS сам определяет тип компьютера, но вы можете также указать его сами, если HIMEM.SYS не может или неправильно определяет тип вашего компьютера. Задайте вместо значений *xxxx* код компьютера или его символьное обозначение, согласно приведенной ниже таблице:

Тип	Код	Название компьютера
at	1	IBM AT or 100% compatible
ps2	2	IBM PS/2
ptlcascade	3	Phoenix Cascade BIOS
hpvectra	4	HP Vectra (A & A+)
att6300plus	5	AT&T 6300 Plus
acer1100	6	Acer 1100
toshiba	7	Toshiba 1600 & 1200XE
wyse	8	Wyse 12.5 Mhz 286
tulip	9	Tulip SX
zenith	10	Zenith ZBIOS
at1	11	IBM PC/AT (alternative delay)
at2	12	IBM PC/AT (alternative delay)
css	12	CSS Labs
at3	13	IBM PC/AT (alternative delay)
philips	13	Philips
fasthp	14	HP Vectra
ibm7552	15	IBM 7552 Industrial Computer
bullmicral	16	Bull Micral 60
dell	17	Dell XBIOS

/SHADOWRAM = ON|OFF — значение ON оставляет в области оперативной памяти данные теневого BIOS (копия BIOS из ПЗУ в ОЗУ). Значение OFF освобождает оперативную память от данных BIOS и увеличивает ее размер на 128 K;

VERBOSE — вывод драйвером HIMEM.SYS на дисплей своего текущего статуса и сообщений об ошибках в процессе загрузки HIMEM.SYS в память. По умолчанию HIMEM.SYS выводит на экран сообщения о своих внутренних установках и создаваемых в системе ошибочных ситуациях. Возможно использование вместо VERBOSE ключа V. Для отключения вывода сообщений нажмите и удерживайте клавишу ALT, до тех пор пока HIMEM.SYS не загрузится в память;

Примеры использования:

Допустим, необходимо организовать поддержку дополнительной памяти и использовать 128 дескрипторов управления блоками. В этом случае вы должны использовать в CONFIG.SYS следующую строку:

```
device = c:\dos\himem.sys /NUMHANDLES = 128
```

(!) INTERLNK.EXE

Перенаправление обмена данными на одном или нескольких дисках, установленных на рабочей станции, или портов принтера на один или несколько дисководов или портов принтера на сетевой диск, в сети типа "клиент - сервер". Драйвер INTERLNK.EXE необходим для использования команд DOS INTERLNK и INTERSVR.

Формат директивы подключения:

device = *диск:\маршрут*\interlink.exe /DRIVES:*n* /NOPRINTER /COM:*n*|*адрес* /LPT:*n*|*адрес* /AUTO /NOSCAN /LOW /BAUD:*число* /V

devicehigh = *диск:\маршрут*\interlink.exe /DRIVES:*n* /NOPRINTER /COM:*n*|*адрес* /LPT:*n*|*адрес* /AUTO /NOSCAN /LOW /BAUD:*число* /V

Параметры:

диск:\маршрут - описание полного пути доступа к файлу INTERLNK.EXE.

Ключи

/DRIVES:*n*	установление числа переназначаемых дисков. По умолчанию *n* равно 3. Если вы укажете *n* равным 0, то INTERLNK переназначит только принтеры;
/NOPRINTER	этот ключ запрещает переустанавливать принтеры, когда идет процесс загрузки INTERLNK в память. По умолчанию INTERLNK переустанавливает все имеющиеся порты принтеров;
/COM:*n*|*адрес*	указание номера *n* или *адреса* последовательного порта, который будет использоваться для обмена данными. Если значение *n* или *адреса* порта опущено, то INTERLNK на компьютере "клиента" осуществит поиск всех имеющихся последовательных портов и будет использовать первый порт для обмена данными с сервером.

Если вы используете ключ /COM и не используете ключ /LPT, то INTERLNK будет осуществлять поиск только последовательных портов. По умолчанию INTERLNK осуществляет поиск всех установленных последовательных и параллельных портов;

/LPT:*n*|*адрес* указание номера *n* или *адреса* параллельного порта, который, будет использоваться для передачи данных. Если вы опускаете значение *n* в ключе /LPT, то INTERLNK на компьютере "клиента" осуществит поиск всех имеющихся параллельных портов и будет использовать первый параллельный порт для работы.

Если вы используете ключ /LPT и не используете ключ /COM, то INTERLNK будет осуществлять поиск только параллельных портов. По умолчанию INTERLNK осуществляет поиск всех установленных последовательных и параллельных портов;

/AUTO загрузка драйвера INTERLNK.EXE в память только в том случае, если "клиент" сумел установить связь с сервером. По умолчанию, INTERLNK загружается в память даже в том случае, если "клиент" не установил связь с сервером;

/NOSCAN загрузка INTERLNK.EXE в память до установления связи между "клиентом" и "сервером". По умолчанию, "клиент" сначала устанавливает связь с "сервером" до загрузки INTERLNK.EXE в память;

/LOW загрузка INTERLNK.EXE в основную память, даже если в этот момент является доступной старшая память. По умолчанию INTERLNK.EXE загружается в область старшей памяти, если она является доступной;

/BAUD:*число* установка максимальной скорости передачи в бодах для последовательного порта. Допустимыми значениями являются следующие 9600, 19200, 38400, 57600, 115200. По умолчанию устанавливается скорость передачи равная 115200;

/V предотвращение конфликтов при работе в сети с системным таймером. Используйте этот ключ в случае соединения компьютеров в сети, посредством последовательного интерфейса, и возможности остановки одного из них, когда INTERLINK переназначает диск или порт принтера.

Примеры использования:

Для обеспечения работы сети, использующей последовательный интерфейс и при условии, что принтеры не должны быть переназначены введите в состав CONFIG.SYS следующую директиву:

```
device = c:\dos\interlnk.exe /COM /NOPRINTER
```

Если INTERLNK использует для работы нестандартный последовательный порт с адресом 3F8h включите в состав CONFIG.SYS следующую строку:

```
device = c:\dos\interlnk.exe /COM:3F8
```

(!) POWER

Снижение потребления энергии компьютером, если устройства или программы работают вхолостую.

Формат директивы подключения:

device = *диск:\маршрут*\power.exe ADV|MAX|REG|MIN STD OFF /LOW

Параметры:

диск:\маршрут - описание полного пути доступа к файлу POWER.EXE.

ADV|MAX|REG|MIN - снижение потребления энергии, когда программы или устройства компьютера простаивают. Значение MAX приводит к максимальному сокращению потребления энергии. Значение REG соответствует установке по умолчанию и позволяет достичь оптимального баланса в потреблении энергии. Значение MIN используется в том случае, если использование значений MAX и REG создает неудобства в работе, связанные с отключением устройств компьютера.

STD - если ваш компьютер поддерживает Расширенную Спецификацию Управления Энергией (Advanced Power Managment - ADM), то параметр STD позволит управлять оборудованием компьютера с помощью драйвера POWER.EXE. Если машина не поддерживает спецификацию ADM, то ключ STD не используется

OFF - отключение управления POWER.EXE сбережением энергии.

Ключи:

/LOW	загрузка POWER.EXE в основную память, даже если в этот момент является доступной старшая память. По умолчанию POWER.EXE загружается в область старшей памяти, если она является доступной.

Примеры использования:

c:\dos\power.exe ADV|MAX

Драйвер POWER.EXE будет использовать режим максимального сбережения электроэнергии.

RAMDRIVE.SYS

Организация в оперативной памяти виртуального (электронного) диска.

Формат директивы подключения:

device = *диск:\маршрут*\ramdrive.sys *объем разм_сект число_файлов* /A/E

devicehigh = *диск:\маршрут*\ramdrive.sys *объем разм_сект число_файлов* /A/E

Параметры:

диск:\маршрут - описание полного пути доступа к файлу RAMDRIVE.SYS.

объем - объем оперативной памяти в K, выделяемой для организации электронного диска. Допустимые значения значения *объема* диска лежат в пределах от 4 до 32767. Если параметр *объем* не указан, то создается электронный диск объемом 64 K.

разм_сект - указание размера сектора, создаваемого электронного диска в байтах. Размер сектора может быть равен 128, 256 или 512 байтам. По умолчанию принимается размер сектора равный 512 байт.

число_файлов - этот параметр определяет число файлов, которое может включать корневой каталог. Допустимые значения этого параметра меняются от 2 до 1024. если этот параметр не указан, то число файлов корневого каталога равно 64.

Ключи

/E	электронный диск создается в дополнительной памяти;
/A	электронный диск размещается в расширенной памяти;

Примеры использования:

device = c:\dos\ramdrive.sys 2048 512 512 /E

В дополнительной памяти создан электронный диск объемом 2048 K, с размером одного сектора 512 байт и числом записей корневого каталога 512.

device = c:\dos\ramdrive.sys 1024 256 128 /A

В расширенной памяти создается электронный диск размером 1024 байт, размером одного сектора 256 байт и числом записей корневого каталога 128.

Примечания:

- Электронный диск по своей организации и работе абсолютно идентичен жесткому диску, но существенная разница между ними заключается в том, что с выключением питания вся информация на электронном диске пропадает. Несмотря на этот недостаток, электронный диск может стать очень эффективным инструментом в вашей работе, т.к. он позволяет значительно увеличить скорость обмена данными между памятью и дисковыми накопителями. Используйте его для работы с часто используемыми файлами, скопировав предварительно файлы с НГМД или винчестера в память.
- В процессе загрузки ОС присваивает электронному диску идентификатор, как и любому другому логическому диску. Следует отметить, что для использования под электронный диск дополнительной памяти, директиве DEVICE = RAMDRIVE.SYS должна предшествовать другая директива файла CONFIG.SYS DEVICE = HIMEM.SYS. В случае использования расширенной памяти под электронный диск следует добавить в файл CONFIG.SYS помимо директивы DEVICE = HIMEM.SYS директиву DEVICE = EMM386.
- Ключи /A и /E несовместимы в директиве DEVICE = RAMDRIVE.SYS, но возможно создание нескольких электронных дисков в старшей памяти различного типа.

(!) SETVER.EXE

Загрузка в память таблицы версий операционной системы.

Формат директивы подключения:

device = *диск:\маршрут*\setver.exe

devicehigh = *диск:\маршрут*\setver.exe

Параметры:

диск:\маршрут - описание полного пути доступа к файлу SETVER.EXE.

Примечание:

Производится загрузка в оперативную память таблицы версий операционной системы. Этот драйвер сообщает прикладной программе версию операционной системе при загрузке ее в память. Если вы исполь-

зуете драйвер, который работает с предыдущими версиями MS-DOS, то необходимо использовать директиву DEVICE = SETVER.EXE до директивы DEVICE=драйвер-устройства в файле CONFIG.SYS.

SMARTDRV.EXE

Организация в дополнительной или расширенной памяти кэша для дисковых накопителей, для ускорения операций ввода-вывода.

Кэш - память функционирует примерно также, как и буферы DOS, однако ее размер может быть существенно больше размера буферов DOS и под дисковый кэш возможно использование дополнительной или расширенной памяти.

Формат директивы подключения:

device = *диск:\маршрут*\smartdrv.exe /DOUBLE_BUFFER

Параметры:

диск:\маршрут - описание полного пути доступа к файлу SMARTDRV.EXE.

Ключи

/DOUBLE_BUFFER	этот ключ предписывает SMARTDRV.EXE организовывать двойную буферизацию операций ввода - вывода данных.

Примеры использования:

device = c:\dos\smartdrv.exe /DOUBLE_BUFFER

В старшей памяти организуется кэш для дисков с двойной буферизацией.

Примечания:

Для организации кэша в дополнительной памяти директиве DEVICE = SMARTDRV.EXE в файле CONFIG.SYS должна предшествовать директива DEVICE = HIMEM.SYS, а при организации кэша в расширенной памяти в CONFIG.SYS должна также быть включена строка DEVICE = EMM386 перед директивой DEVICE = SMARTDRV.EXE.

Для организации загрузки SMARTDRV в память и установки режимов его работы вы также можете использовать файл AUTOEXEC.BAT, включив следующую строку SMARTDRV параметры ключи. Подробности управления драйвером SMARTDRV рассмотрены в разделе "Команды DOS" в описании команды SMARTDRV.

Модуль, который осуществляет двойную буферизацию в составе SMARTDRV.EXE, загружается в основную память. Он использует 2 K основной памяти и не может быть загружен в старшую память.

Двойная буферизация особенно эффективна при работе с винчестерами, использующими интерфейс SCSI (Small Computer System Interface - интерфейс малых компьютерных систем) и системную шину EISA, а также при использовании системной шины MCA (MicroChannel Architecture - архитектура "микроканал").

Используйте также двойную буферизацию при работе с EMM386 и при работе с Windows в расширенном режиме 386 процессора.

Если вы не знаете, как правильно установить параметры SMARTDRV, выполните следующие шаги:

1. Подключите SMARTDRV к ОС директивой DEVICE = SMARTDRV.EXE;
2. Используйте в AUTOEXEC.BAT команду SMARTDRV;
3. Запустите утилиту MemMaker;
4. Используйте команду MEM /C /P для получения информации об использовании оперативной памяти компьютера;
5. Наберите в командной строке DOS "SMARTDRV" и вы получите информацию о текущем состоянии драйвера SMARTDRV.EXE

8 Команды DOS

Команды DOS, как было отмечено выше, бывают двух типов:

- Внутренние (команды, встроенные в COMMAND.COM).
- Внешние (утилиты) - команды, выполненные в виде отдельных исполняемых файлов.

В описании формата внутренних команд мы будем пользоваться следующей схемой:

имя_команды *параметры* / КЛЮЧИ

При описании формата внешних команд будет использована следующая схема:

*диск:**маршрут***имя_команды** *параметры* / КЛЮЧИ

При указании в формате команды текущего диска параметры *диск:**маршрут* использоваться не будут. Для получения справочной информации о команде используйте ключ /?, что приводит к выводу справочной информации о команде, но выполнения команды не происходит.

APPEND

Задание порядка поиска файлов в каталогах и вывод на дисплей порядка поиска файлов, установленного ранее командой APPEND. Команда внешняя. Файл APPEND.EXE

Формат команды:

*диск:**маршрут***append** *диск1:**маршрут1;диск2:**маршрут2;...* /E /X:ON|OFF /PATH:ON|OFF

Параметры:

*диск1:**маршрут1;диск2:**маршрут2;...* - пути поиска требуемых файлов.

Эта команда предписывает DOS, в каких каталогах осуществлять поиск файлов, требуемых прикладным программам.

Ключи:

/X: ON|OFF	использование ключа /X:ON или /X заставляет DOS искать программы, которые используют функции поиска файлов и запускают дочерние процессы. Этот же ключ предписывает включать DOS в сферу поиска выполняемые файлы с расширением .EXE и .COM. Если вы используете ключ /X:OFF, то отменяете действие ключа /X или /X:ON. Не рекомендуется использовать команду APPEND в файле AUTOEXEC.BAT. Лучше применять ее после того, как DOS загрузится. По умолчанию ключ /X установлен в положение OFF.
/PATH: ON|OFF	в процесс поиска включаются все файлы, обрабатываемые командой APPEND. По умолчанию ключ /PATH установлен в положение ON.
/E	пути, перечисленные в команде APPEND заносятся в системное окружение с именем APPEND. Этот ключ можно использовать только один раз при первом выполнении команды APPEND. Если вы используете ключ /E, вы также можете использовать команду SET для вывода на дисплей установленных переменных системного окружения. Дополнительная информация о системном окружении рассмотрена в главе "Системное окружение", а также в описании команды SET.

Примечания:

Команду APPEND нельзя использовать при использовании вами Microsoft Windows или Windows Setup. Подобности поиска командой APPEND выполняемых файлов приведены в описании команды PATH.

Примеры использования:

Для организации поиска файлов в каталоге LETTERS на диске B и в каталоге REPORTS диска A используйте следующую команду:

```
append b:\letters; a:\reports
```

Для включения в процесс поиска файлов, требующихся программам для организации процессов поиска файлов и запуска дочерних процессов введите команду

```
append /E
```

Эта же команда будет предписывать DOS осуществлять поиск файлов с расширением .EXE и .COM.

ATTRIB

Изменение атрибутов файлов и каталогов (кроме корневого), а также вывод текущих атрибутов файла (каталога) на экран. Команда внешняя. Файл ATTRIB.EXE.

Формат команды:

attrib +R|-R +A|-A +S|-S +H|-H *диск:\маршрут\имя-файла* /S

Параметры:

диск:\маршрут - описание полного пути доступа к файлу ATTRIB.EXE.

+R присвоение файлу атрибута "только для чтения", который запрещает изменение и удаление указанного файла.

-R изменение для файла атрибута "только для чтения". Файл может быть подвергнут изменениям или удалению.

+A присвоение файлу атрибута "архивный". Эта операция выполняется для того, чтобы объявить файл вновь созданным или подвергшимся архивации. Обычно DOS сама устанавливает флаг архивации после первой же модификации файла. Чаще всего информация об атрибуте файла "архивный" используется программами резервного копирования, например утилитой DOS BACKUP. Эта же информация используется утилитой XCOPY.

-A сброс атрибута "архивный". Эту операцию обычно выполняет команда BACKUP, а также специализированные программы - архиваторы (ARJ, PKZIP, PKARC и др.).

+S установка атрибута "системный". Этот флаг почти всегда установлен для файлов IO.SYS и MSDOS.SYS.

-S снятие атрибута "системный".

+H установка для файла атрибута "скрытый". "Скрытые" файлы не показываются командой DIR, но выводятся на экран в списке файлов некоторыми утилитами или программами - оболочками DOS, например Norton Commander. К "скрытым" файлам относятся файлы IO.SYS и MSDOS.SYS.

-H с файла снимается атрибут "скрытый".

Ключи

/S	процесс изменения атрибутов файла(ов) распространяется не только на текущий каталог, но и на все каталоги, подчиненные текущему.

Примеры:

Для вывода информации об атрибутах файла введите команду:

```
attrib io.sys
```

Изменить атрибуты файла PRIMER.TXT можно, если ввести следующую команду:

```
attrib primer.txt +A +H +R
```

В результате выполнения вышеприведенной команды файлу PRIMER.TXT будут присвоены следующие атрибуты:

- "архивный";
- "скрытый";
- "только для чтения".

Получив эти атрибуты, файл primer.txt будет защищен от модификации и удаления, не будет показываться на экране командой DIR и будет сохранен при резервном копировании, например командой BACKUP.

Изменять атрибуты можно также и у каталогов. Для примера, можно изменить атрибуты подкаталога SYSTEM и всех подчиненных ему, введя следующую команду:

```
attrib +H c:\system /S
```

В команде ATTRIB допускается использование групповых шаблонов в имени файла. Например для снятия атрибута "только для чтения" во всех файлах всех каталогов диска A введите команду:

```
attrib -R a:\*.*/S
```

Для объявления всех файлов с расширением .EXE "скрытыми" используйте следующую команду:

```
attrib +H *.exe
```

Примечания:

- В одной команде ATTRIB можно указывать несколько параметров для установки атрибутов файлов. При использовании нескольких параметров в одной команде они должны разделяться пробелами.
- Команда ATTRIB без параметров выводит текущие атрибуты указанного файла.

BREAK

Управление реакцией системы на нажатие клавиш Ctrl-C или Ctrl-Break. Команда внутренняя.

Формат команды:

break ON|OFF

Параметры:

ON|OFF - при задании этого ключа DOS будет принудительно прекращать выполнение программы при нажатии комбинации клавиш Ctrl-C или Ctrl-Break. Эти команды анализируются системой только в моменты организации операций ввода-вывода данных, а не в моменты нажатия клавиш на клавиатуре.

Если в команде BREAK установлен параметр ON, то контроль нажатия клавиш Ctrl-C или Ctrl-Break организуется перед началом всех операций ввода-вывода.

Если в команде BREAK установлен параметр OFF, то контроль нажатия клавиш Ctrl-C или Ctrl-Break выполняется только перед операциями работы с дисплеем, клавиатурой, параллельными и последовательными портами.

Примечания:

- Команда BREAK без параметров выводит свою текущую установку.
- Использование команд Ctrl-C или Ctrl-Break неэффективно в тех случаях, когда выполняемая программа редко использует операции ввода-вывода или совсем их не использует.
- Команда BREAK ON|OFF эквивалентна использованию директивы файла CONFIG.SYS break ON|OFF. По умолчанию для команды BREAK установлен параметр OFF.

CHCP

Смена текущей кодовой таблицы. Команда внутренняя.

Формат команды:

chcp *nnn*

Параметры:

nnn - номер, устанавливаемой кодовой таблицы. Команда производит замену текущей кодовой таблицы (набор символов) на кодовую таблицу с номером *nnn*. Параметр *nnn* представляет номер кодовой таблицы в диапазоне от 0 до 999. Дополнительную информацию о кодовых таблицах можно найти в описании команды KEYB.

Примеры использования:

Для получения информации о текущей кодовой странице, введите команду:

chcp

В ответ MS-DOS выведет следующее сообщение:

Active code page: 437

Для установки другой кодовой таблицы, например 850 (многоязычной), введите команду:

chcp 850

В ответ MS-DOS попытается установить выбранную кодовую таблицу. Если она ее не обнаружит, то на дисплее вы увидите следующее сообщение:

Invalid code page

Если устройство (монитор, клавиатура, принтер) не поддерживает указанную кодовую таблицу, DOS выведет на дисплей сообщение об ошибке следующего формата:

Code page 850 not prepared for device nnn

Примечания:

- Перед тем как использовать команду, вы должны указать местонахождение на диске файла COUNTRY.SYS в директиве файла CONFIG.SYS и загрузить в память утилиту NLSFUNC.
- Если в файл CONFIG.SYS включена директива DEVICE = DISPLAY.SYS, то команда CHCP загружает для дисплея указанную кодовую таблицу.
- Команда CHCP изменяет кодовые таблицы для всех устройств (дисплея, принтера), в отличии от команды MODE, которая позволяет изменить кодовую таблицу только для одного устройства.

CHDIR (CD)

Изменение текущего каталога или вывод на дисплей текущего каталога диска. Команда внутренняя.

Формат команды:

chdir *диск:\маршрут*
cd *диск:\маршрут*

Параметры:

диск:\маршрут - указание полного пути к каталогу, назначаемому текущим. Если диск не указан, то считается, что назначаемый каталог находится на текущем диске.

Примеры использования:

```
cd \primer
chdir \primer
```

Осуществляется переход из текущего каталога в подчиненный ему каталог PRIMER. Для указания каталога, который необходимо сделать текущим, можно использовать непосредственное задание имени каталога. Нижеприведенный пример иллюстрирует этот метод указания каталога, который необходимо сделать текущим:

```
cd \specials\sponsors
```

Из текущего каталога осуществляется переход в подчиненный каталог SPECIALS, а из подкаталога SPECIALS в подкаталог SPONSORS. Перейти в подкаталог SPONSORS можно и использовав следующий метод:

```
cd \specials
cd \sponsors
```

Для возвращения из подчиненного каталога в родительский используется следующая команда:

```
cd .
```

Для вывода на дисплей имени текущего каталога используйте команду CHDIR или CD без параметров. Допустим вам необходимо посмотреть имя текущего каталога. Пусть им является каталог JONES, подчиненный каталогу PUBLIC диска B:. После ввода команды CHDIR на экране дисплея вы увидите следующее сообщение:

```
b:\public\jones
```

Если вы работаете на диске D:, и вам необходимо скопировать все файлы из каталогов \PUBLIC\JONES и \PUBLIC\LEWIS диска C: в корневой каталог диска D: наберите следующие команды:

```
chdir c:\public\jones
copy c:*.* d:\
chdir c:\public\lewis
copy c:*.* d:\
```

Если вы желаете скопировать все файлы из каталогов \PUBLIC\JONES и \PUBLIC\LEWIS диска C: в ваш текущий каталог на диске D:, наберите следующие команды :

```
chdir c:\public\jones
copy c:*.* d:
chdir c:\public\lewis
copy c:*.* d:
```

В данном примере используется команда COPY, подробную информацию о которой можно найти в описании этой команды в этой главе.

Команда CHDIR (CD) позволяет переходить из подчиненного каталога в другие подчиненные каталоги, используя переходы в каталоги более высокого уровня, а затем в необходимый подчиненный каталог. Приведенный ниже пример показывает, как осуществить такой переход:

```
C:\LETTERS\JUL>cd ..\..\notes\work
```

В приведенном примере осуществляется переход из каталога JUL сначала в родительский каталог LETTERS, из каталога LETTERS в родительский для LETTERS корневой каталог диска C. Далее будет осуществлен переход из корневого каталога диска C в подчиненный ему каталог NOTES, а из каталога NOTES в каталог WORK.

Примечания:

- Команда CHDIR (CD) не позволяет переходить из подчиненного каталога в родительский используя команду CD имя-родит.кат.
- Для перехода в родительский каталог используйте команду CD .., а для перехода в корневой CD \.
- Смена текущего каталога на указанном в команде CD диске (отличающемся от текущего) не изменяет текущий диск.

CHKDSK

Проверка логического диска (файла, каталога) на правильность организации файловой структуры. Команда внешняя. Файл CHKDSK.EXE.

Формат команды:

chkdsk *диск:\маршрут\имя-файла* /F /V

Параметры:

диск:\маршрут - указание полного пути файлу (файлам), предназначенным для проверки.

имя-файла - имя файла, назначенного для проверки. При указании имен файлов допускается использовать групповые шаблоны * и ?. Если указаны шаблоны * и ?, то проверке подвергаются все файлы, удовлетворяющие заданным шаблонам.

Ключи

/F производится фиксация обнаруженных ошибок. Этот ключ нельзя использовать при запуске команды CHKDSK из среды Microsoft Windows или Microsoft Task Swapper. Если ключ /F

не используется, то CHKDSK будет только обнаруживать ошибки, но не исправлять их;

/V при использовании этого ключа CHKDSK будет выводить на дисплей имя каждого проверяемого файла в каждом проверяемом каталоге;

Примеры использования:

chkdsk a: /F

Организуется проверка диска A на правильность организации его файловой структуры, одновременно будет производиться исправление выявленных ошибок организации файловой структуры.

chkdsk a: /F > status

При выполнении этой команды будет организована проверка диска A, аналогично команде CHKDSK A:, но отчет о результатах работы команды CHKDSK будет выведен не на дисплей, а в файл STATUS. Для получения информации о текущем диске нужно просто ввести команду

chkdsk

После выполнения этой команды DOS выведет на дисплей следующий отчет:

```
Volume Serial Number is B2AF-AFBF

72214528 bytes total disk space
   73728 bytes in 3 hidden files
   30720 bytes in 12 directories
11493376 bytes in 386 user files
   61440 bytes in bad sectors
60555264 bytes avaliable on dusk

    2048 bytes in each allocation unit
   35261 total allocation unit on disk
   29568 avaliable allocatin unit on disk

  655360 total bytes memory
  493456 bytes free
```

Приведенный выше отчет содержит в себе сведения об объеме диска в байтах, объеме занимаемом "скрытыми" файлами, каталогами и файлами пользователя, а также объеме диска, занятом плохими секторами и количестве свободного дискового пространства.

Далее приведена информация о размерах 1 кластера, общем количестве и количестве доступных пользователю кластеров. Последний раздел этого отчета содержит информацию об объеме основной оперативной памяти и об объеме памяти, доступной пользователю.

Примечания:

- Если CHKDSK используется с ключом /F, то при обнаружении "висячих" кластеров на экран будет выведено сообщение:

  10 (nnn) lost allocation units found in 3 chains
  Convert lost chains to file ?
  "10 потерянных кластеров найдено в 3 цепочках"
  "Преобразовать цепочки в файл ?"

  Если вы нажмите клавишу Y в ответ на запрос DOS, то CHKDSK преобразует потерянные кластеры в файл(ы) формата FILEnnnn.CHK. После окончания работы утилиты CHKDSK вы можете посмотреть информацию в файлах с расширением .CHK. Если в ответ на запрос DOS вы введете N, то CHKDSK исправит ошибку, но не будет создавать файл(ы) с расширением .CHK.

- Утилита CHKDSK не может работать с каталогами над которыми осуществлялись операции с использованием команды SUBST.

- В процессе своей работы CHKDSK также обнаруживает файлы, которые имеют "пересекающиеся" кластеры. Эти кластеры возникают в тех случаях, когда из-за ошибок DOS 2 и более файлов используют одни и те же кластеры. При обнаружении таких кластеров CHKDSK выведет на дисплей сообщение:

  (File) is cross-linked on allocation unit (number)
  (Файл) содержит "пересекающийся" кластер (номер)

  Если вы используете ключ /F, то CHKDSK не производит обработку ошибок, связанных с "пересекающимися" кластерами. Если ключ /F указан, то производится устранение ошибок связанных с "пересекающимися" кластерами.

- По окончании своей работы CHKDSK возвращает значение ERRORLEVEL, равное 0, если ошибок организации файловой структуры не обнаружено, и равное 255, если такие ошибки есть.

CLS

Очистка экрана. Команда внутренняя.

Формат команды:

cls

Примечание:

Команда очищает экран и устанавливает курсор в верхнюю левую позицию.

COMMAND

Команда внешняя. Файл COMMAND.COM.

Вызов на выполнение второго командного процессора. Получив эту команду, DOS загружает в память копию командного процессора с его окружением. Для выхода из второго командного процессора используется команда EXIT. Выполнив команду EXIT, DOS возвращает управление первому командному процессору.

Формат команды:

command *диск:\маршрут устройство* /C *строка* /E:*nnnnn* /K *имя-файла* /P /MSG

Параметры:

диск:\маршрут - указание местоположения второго командного процессора. Этот параметр изменяет переменную системного окружения COMSPEC. Второй командный процессор может находиться в любом каталоге любого диска, в отличие от первого командного процессора, который всегда должен находиться в корневом каталоге системного диска.

устройство - задание системного устройства ввода-вывода для ввода и вывода команд DOS. По умолчанию этим устройством является консоль (CON).

Ключи

/C *строка*	задание строки, которая при запуске передается в качестве параметра второму командному процессору;
/E:*nnnnn*	задание размера системного окружения в байтах. Размер системного окружения меняется в пределах от 160 до 32768 байт ступенями по 16 байт. По умолчанию размер системного окружения установлен равным 256 байт;
/K *имя-файла*	запуск программы или командного файла из командной строки DOS при активизации второго командного процессора. Этот ключ обычно используется для запуска файла, подобного AUTOEXEC.BAT. в среде Windows. Для детального изучения этого вопроса просмотрите в среде Windows программой Pif Editor файл DOSPRMT.PIF. Не рекомендуется использовать ключ /K в директиве SHELL файла CONFIG.SYS. Это может привести к проблемам работы прикладных программ, запускаемых в процессе обработки файла AUTOEXEC.BAT;

/P — при использовании этого ключа из загруженного командного процессора нельзя выйти командой EXIT, и он остается резидентным в памяти;

/MSG — запись и сохранение в памяти текстов аварийных сообщений. Этот ключ обычно используется при работе с гибкими дисками. Без этого ключа тексты аварийных сообщений считываются с диска. Для использования ключа /MSG вы также должны использовать в команде COMMAND и ключ /P.

Примеры использования:

```
command
```

Загрузка в оперативную память второго командного процессора.

```
command /C chkdsk
```

Загрузка в память второго командного процессора с последующим выполнением файла CHKDSK.EXE.

```
command /C runme.bat
```

Выполнение из одного командного файла другого, с возвратом управления первому командному файлу после завершения второго. Аналогично использованию команды CALL пакетных файлов.

Примечания:

- Когда в память загружается новый командный процессор, MS-DOS создает для него системное окружение, которое полностью идентично системному окружению первого командного процессора. Вы можете изменить сами размер системного окружения, использовав команду COMMAND с ключом /E:*nnnnn*;
- Если ОС загружена в область старшей памяти, то в эту же область будет загружена и резидентная часть командного процессора для увеличения размера основной памяти для прикладных программ.

COPY

Копирование и объединение файлов. Команда внутренняя.

Формат команды:

Для копирования файлов:

copy /A /B *диск-источник:\маршрут1\имя_файла(ов)* /A /B *диск-приемник:\маршрут2* /A /B /V

Для объединения файлов:

copy /A /B *диск1:\маршрут1\имя_файла1* /A /B +*диск2:\маршрут2\имя_файла2* /A /B+...*дискN:\маршрутN\имя_файлаN* /A /B /V

Параметры:

Для копирования файлов:

диск-источник:\маршрут1\имя_файла(ов) - указание полной спецификации файла(ов), подлежащих копированию.

диск-приемник:\маршрут - указание спецификации диска и каталога, куда будет произведено копирование файлов.

Для объединения файлов:

диск1:\маршрут1\имя_файла1 - указание полной спецификации 1-го файла, подлежащего объединению.

диск2:\маршрут2\имя_файла2 - указание полной спецификации 2-го файла, подлежащего объединению.

дискN:\маршрутN\имя_файлаN - указание полной спецификации N-го файла, подлежащего объединению.

Ключи

/A — при использовании этого ключа в команде COPY входные файлы рассматриваются как текстовые, т.е. состоящие из последовательности кодов ASCII. В этом случае копируется содержимое файла, пока не встретится код 26 (десятичное) - конец файла (EOF). Сам символ конца файла в этом случае не копируется.

Выходные файлы при использовании ключа /A также рассматриваются, как текстовые.

После завершения копирования к концу файла добавляется код 26. Ключ /A действует на все последующие файлы, в том числе и на предшествующий, до тех пор, пока не встретится файл с ключом /B;

/B — если в команде COPY установлен этот ключ, то входные файлы рассматриваются как последовательность битов, и количество копируемых байтов зависит только от длины самого файла. Символы конца файла копируются так же, как и все другие символы. Выходные файлы с этим ключом также рассматриваются, как двоичные. При этом после окончания копирования код 26 в конец файла не записывается. Ключ действует на все последующие файлы, в том числе и на предшествующий ему файл, до тех пор, пока не встретится файл с ключом /A;

/V включение проверки правильности копирования. При использовании этого ключа после окончания копирования производится проверка правильности его записи.

Примеры использования:

Для копирования файла в другой файл и записи в его конец кода конца файла EOF используйте следующую команду:

```
copy memo.doc letter.doc /A
```

Если вам необходимо скопировать файл из текущего каталога в другой, вы можете не указывать в команде COPY местонахождение копируемого файла. Приведенный ниже пример показывает, как скопировать из текущего каталога файл ROBIN.TYP в каталог BIRDS диска C:

```
copy robin.typ c:\birds
```

Если каталога BIRDS не существует, DOS скопирует файл ROBIN.TYP в файл с именем BIRDS и поместит его в корневой каталог диска C:.

Для объединения нескольких файлов в один можно также использовать команду COPY. Имена файлов, подлежащих объединению, должны быть перечислены в команде COPY через символ "+". Следующий пример показывает, как объединить в один файл несколько других файлов:

```
copy mar89.rpt + apr89.rpt + may89.rpt report
```

В результате выполнения этой команды файлы с именами MAR89.RPT, APR89.RPT и MAY89.RPT, находящиеся в текущем каталоге, будут объединены в один файл с именем REPORT, который будет помещен также в текущий каталог. Когда происходит процесс объединения файлов, создаваемый объединенный файл будет иметь текущую дату и время создания.

Если вам необходимо добавить в созданный файл данные из других файлов, вы можете это сделать, как показано ниже. В приводимом примере файл REPORT, который уже существует, объединяется с другими 4 файлами в новый файл с тем же именем REPORT:

```
copy report + mar89.rpt + apr89.rpt + may89.rpt
```

При использовании команды COPY вы можете также использовать и шаблоны групповых операций:

```
copy *.txt combin.doc
```

Приведенная выше команда предписывает DOS скопировать все файла текущего каталога текущего диска, имеющие расширение .TXT, в файл с именем COMBIN.DOC и поместить его в текущий каталог теку-

щего диска. Для объединения различных двоичных файлов в один используйте ключ /B, как показано в следующем примере:

```
copy /B *.exe combin.exe
```

После окончания операции объединения DOS попросит вас ввести символ конца файла. Чтобы выполнить эту операцию, нажмите клавишу F6 или комбинацию клавиш Ctrl+Z и затем ввод.

Предупреждение: *если вы объединяете двоичные файлы, то полученный файл не обязательно будет иметь тот же формат, что и входные файлы.*

В следующем примере команда COPY комбинирует каждый файл с расширением .TXT с каждым файлом, имеющим расширение .REF и результат записывает в файл с расширением .DOC. Предполагается, что все объединяемые файлы находятся в текущем каталоге текущего диска. Файлы, полученные в результате выполнения этой команды COPY, будут также находится в текущем каталоге текущего диска.

В результате выполнения этой команды, например, файл FILE1.TXT будет объединен с файлом FILE1.REF, а результат объединения будет записан в файл с именем FILE1.DOC:

```
copy *.txt + *.ref *.doc
```

Следующий пример иллюстрирует использование команды COPY для объединения всех файлов текущего каталога с расширениями .TXT и .REF в файл COMBIN.DOC:

```
copy *.txt + *.ref combin.doc
```

Команда COPY может быть также использована для организации простейшего метода создания файлов. Допустим, необходимо создать файл OUTPUT.TXT набором его с клавиатуры. Это можно сделать, введя следующую команду:

```
copy con output.txt
```

После того, как вы наберете эту команду и нажмете клавишу ENTER, MS-DOS приступит к копированию символов, вводимых с клавиатуры в файл OUTPUT.TXT. Когда набор файла закончен, нажмите клавиши Ctrl-Z или F6 и далее клавишу Enter. После этого DOS скопирует файл и сформирует сообщение:

```
1 file(s) copied
```

которое информирует пользователя о том, что 1 файл скопирован. Созданный файл будет находиться в текущем каталоге текущего диска.

Команда COPY может также использоваться и для изменения даты и времени создания файла. Для изменения даты и времени создания

файла на текущие необходимо ввести команду COPY в следующем формате:

```
copy /B source+,,
```

В указанном примере изменяются время и дата создания файла SOURCE на текущие. Символ ",,", стоящий после символа "+", указывает DOS, что список входных файлов исчерпан.

```
copy system.bat+con
```

Содержимое файла SYSTEM.BAT объединяется с символами, вводимыми с клавиатуры.

```
copy con+system.bat system.bat
```

Строки символов, вводимые с клавиатуры добавляются в начало уже существующего файла SYSTEM.BAT.

Примечания:

- При использовании команды COPY в качестве входной и выходной спецификации файлов допускается указание устройств, например AUX, CON, COM1, LPT.
- Если в выходной спецификации отсутствует имя файла, то в качестве выходных имен файлов принимаются имена входных файлов.
- Если при указании спецификации отсутствует описание диска и каталога, то в качестве идентификаторов диска и каталога подразумеваются текущий каталог текущего диска.
- По умолчанию для файлов, копируемых с диска на диск в команде COPY установлен ключ /B, т.е. файлы рассматриваются, как двоичные.
- В команде указывается сначала входная спецификация, а затем выходная.
- Если входная спецификация содержит шаблоны групповых операций, а выходная спецификация однозначно указывает на имя файла, то после выполнения команды COPY выходной файл будет образован путем объединения файлов, удовлетворяющих заданным шаблонам.
- При организации копирования с одним указанием спецификации выходной спецификацией считается текущий диск и текущий каталог, выходным файлам присваиваются имена входных.
- При операциях копирования с указанием только входной спецификации рабочие диск, путь или оба приведенных параметра должны отличаться от указанных (т.е. файл нельзя скопировать "сам на

себя"). Кроме того, в этом случае во входной спецификации нельзя указывать устройство.

- Применительно к операциям объединения выходным считается последний файл из списка, если перед ним не имеется символа "+". Если такого файла не имеется, то выходным считается первый из списка входных файлов.
- В операциях объединения не рекомендуется указывать несколько выходных спецификаций. В случае если же все-таки используется шаблон групповой операции, то в результате выполнения команды COPY будет создано несколько выходных файлов.
- Объединяемые файлы считаются текстовыми. Для объединения двоичных файлов используйте ключ /B.

CTTY

Изменение системного устройства управления компьютером. Команда внутренняя.

Формат команды:

ctty *устройство*

Параметры:

устройство - в качестве указанного устройства в системе назначается новое стандартное устройство ввода-вывода информации. В качестве таких устройств допускается использование следующих: PRN, LPT1, LPT2, LPT3, CON, AUX, COM1, COM2, COM3, COM4.

Примеры использования:

Для изменения устройства, с которого будет осуществляться ввод команд и вывод результатов их выполнения (таковым по умолчанию является консоль CON) на последовательный порт AUX, введите команду:

```
ctty aux
```

После выполнения этой команды системным устройством ввода - вывода станет первый последовательный порт AUX (он же COM1).

Для возвращения управлением DOS с консоли, введите нижеприведенную команду:

```
ctty con
```

Примечания:

- Перед тем, как установить командой CTTY системным устройством ввода-вывода последовательный порт, желательно установите его характеристики командой MODE (скорость передачи в бодах, паритет, число информационных и стоповых битов). Порядок установки параметров последовательного порта дан в описании команды MODE;
- Желательно использование в качестве стандартного устройства ввода-вывода, устройства, которое может передавать информацию в обе стороны. Некоторые устройства не позволяют организовать обмен информацией в обе стороны, например принтер или "пустое" устройство - NUL.
- Команда CTTY, изменяя стандартное устройство ввода-вывода информации, не меняет организацию других процессов перенаправления ввода-вывода и конвейеры (<, >, >>, |).

DATE

Установка или вывод на дисплей текущей даты. Команда внутренняя.

Формат команды:

date *дата*

Параметры:

дата - установка даты в системе. Параметр "дата" может вводиться в одном из следующих форматов:

мм-дд-уу или *мм-дд-уууу* (стандарт США)

дд-мм-уу или *дд-мм-уууу* (европейский стандарт)

уу-мм-дд или *уууу-мм-дд* (азиатский стандарт)

Значения *мм* меняется от 1 до 12. Значения *дд* изменяются в пределах от 1 до 31. Значения *уу* или *уууу* изменяются от 80 до 99 или от 1980 до 1999 соответственно.

Примеры использования:

date

На экран дисплея DOS выведет системную дату с возможностью его замены.

date 25-06-89

Устанавливается системная дата 25 июля 1989 года.

date 04/12/92

Устанавливается системная дата 4 декабря 1992 года.

Примечания:

- Вид представления текущей даты определяется установкой кода страны в файле CONFIG.SYS.
- Разделителями в команде DATE могут служить символы "-" и "/".
- Кроме текущей даты, DOS также выводит на дисплей и сокращенное обозначение дня недели. Обозначения дней недели приведены ниже:

Mon	-	понедельник	Fri	-	пятница
Tue	-	вторник	Sat	-	суббота
Wed	-	среда	Sun	-	воскресенье
Thu	-	четверг			

(!) DBLSPACE

Сжатие данных на ваших жестких или гибких дисках и создание дополнительных дисков, работающих под управлением программы Double Space.

Создание дополнительных дисков организуется путем создания специальных файлов "сжатых" томов (Compressed Volume File - CVF), которые в дальнейшем рассматриваются, как логические диски и работа с которыми аналогична работе с другими "несжатыми" дисками.

Когда вы вводите команду DBLSPACE, DOS запускает на выполнение программу Double Space. Эта программа имеет удобный пользовательский интерфейс и очень проста в работе. Пользователю только необходимо отвечать на вопросы, задаваемые DOS.

Кроме этого, пользователю предоставляется в распоряжение хорошо продуманная система помощи и подсказок, позволяющая делать как можно меньше ошибок при работе с Double Space.

Внешний вид экрана при запуске Double Space приведен ниже на рисунке 8.1

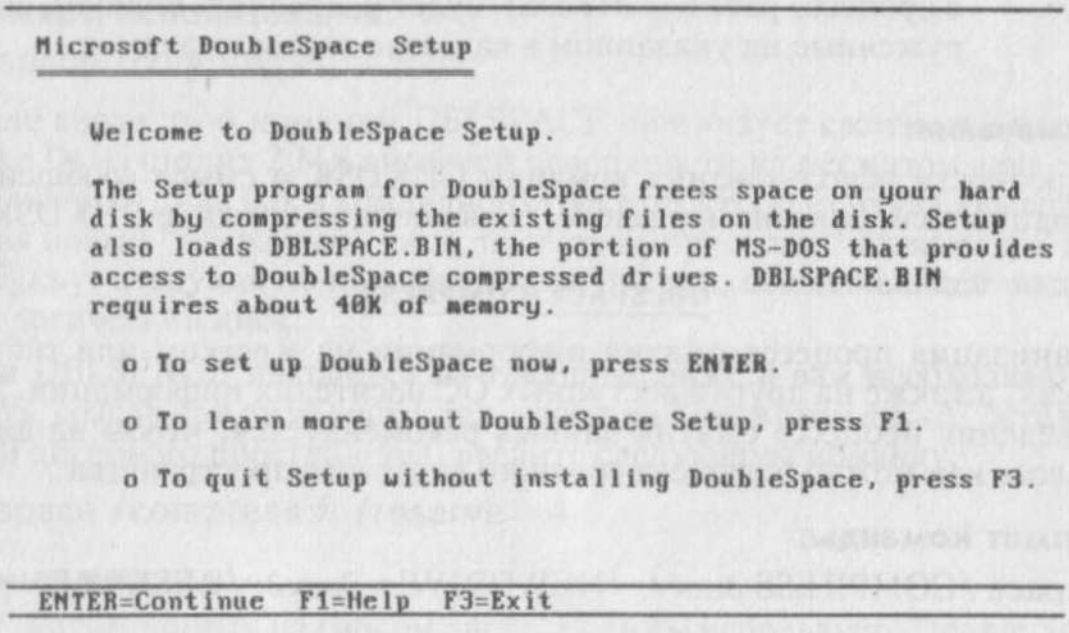


Рис 8.1

Работа с программой Double Space может быть также организована с использованием командной строки DOS. При задании режимов работы из командной строки DBLSPACE будет выполнять предписанные ей действия от задачи к задаче без участия пользователя.

Режимы работы DBLSPACE описаны ниже.

DBLSPACE/CHKDSK

Проверка правильности организации файловой структуры выбранного диска. В процессе работы в этом режиме DBLSPACE запускает утилиту CHKDSK, которая берет на себя проверку корневого каталога, FAT на правильность записи файлов.

Формат команды:

dblspace /CHKDSK /F *диск:*

Параметр:

диск: - указание буквы, обозначающей проверяемый CHKDSK логический диск. Если этот параметр опущен в команде, Double Space проверяет текущий диск.

Ключи

/CHKDSK	проверка файловой структуры диска. Допускается использование вместо ключа /CHKDSK аббревиатуры /CHK;

/F — в процессе работы CHKDSK будет исправлять ошибки, обнаруженные на указанном в качестве параметра диске.

Примечание:

Подробности использования команды CHKDSK и смысл сообщений, выводимых ей, описаны в разделе, посвященном команде CHKDSK.

DBLSPACE/COMPRESS

Организация процесса сжатия информации на жестком или гибком дисках, а также на других доступных ОС носителях информации. Для проведения процесса сжатия данных рекомендуется, чтобы на диске имелось как можно больше свободного дискового пространства.

Формат команды:

dblspace /COMPRESS *диск1:* /NEWDRIVE=*диск2:* /RESERVE=*размер*

Параметр:

диск1: - указанная буква определяет логический диск, который подлежит сжатию.

Ключи

/COMPRESS	сжатие данных на жестком или гибком диске, указанном в параметре диск1;
/NEWDRIVE=*диск2*	указание буквы диска, которая будет присвоена "несжатому" - "горячему" диску после процесса сжатия данных. После сжатия данных на указанном ваша система будет включать в себя два логических диска: один "сжатый" диск и новый диск, не подвергшийся сжатию данных. При отсутствии этого ключа ОС назначает "горячему" диску букву, следующую по алфавиту за буквой, обозначающей последний, доступный ОС логический диск. Этот ключ можно задавать в следующем виде: /NEW;
/RESERVE=*размер*	задание размера дискового пространства в мегабайтах, не подлежащего сжатию. Этот ключ необходимо использовать в тех случаях, когда вы работаете с Windows. В процессе своей работы Windows создает временные файлы, с которыми нельзя организовать работу на сжатом диске. Этот ключ допустимо обозначать аббревиатурой /RES.

Примеры использования:

dblspace /compress d:

После ввода этой команды DBLSPACE организует сжатие данных на диске D: и оставит 2 MB дисковой поверхности на несжатом диске (по умолчанию). Так как в командной строке отсутствует буква, обозначающая новый "несжатый" диск, то система присвоит "несжатому" диску букву, следующую по алфавиту за буквой, обозначающей последний логический диск.

Если DBLSPACE находится на текущем диске и вам необходимо его сжать, создав при этом диск F так, чтобы на "несжатом диске" оставить 4 MB дискового пространства, введите следующую команду:

dblspace /compress f: /reserve = 4

Примечания:

- Сжатие данных на гибком диске. Если вы используете Double Space для увеличения емкости вашей дискеты помните, что сможете прочитать ваши сжатые данные или скопировать их с компьютера на компьютер только при наличии на вашем и других компьютерах ОС MS-DOS 6.0 с подключенной утилитой DBLSPACE.
- Double Space не может организовать процесс сжатия ваших данных в том случае, если ваш винчестер или дискета почти или совсем заполнены. Для работы Double Space требуется не менее 1.2 MB свободного дискового пространства на системном диске и не менее 0.65 MB на остальных дисках.
- Утилита DBLSPACE не работает с 360 K дискетами.

DBLSPACE/CREATE

Создание нового сжатого диска с использованием свободного пространства на "несжатом диске".

Формат команды:

dblspace /CREATE *диск1:* /NEWDRIVE = *диск2:* /SIZE = *размер* /RESERVE = *размер*

Параметры:

диск1 - буква, обозначающая логический диск, свободная поверхность которого будет использована для создания "сжатого" диска.

Ключи

/CREATE	создание нового "несжатого" диска, свободная поверхность которого будет использована для созда-

ния "сжатого" диска. Этот ключ можно использовать в сокращенном виде, задав /CR;

/NEWDRIVE = *диск2:* — указание буквы, которая будет присвоена новому логическому "несжатому" диску после окончания работы утилиты DBLSPACE. Если этот параметр опущен, то "несжатому" диску будет присвоена буква, следующая по алфавиту за буквой, обозначающей последний логический диск системы. Допустимо вместо ключа /NEWDRIVE использовать ключ /N;

/RESERVE = *размер* — задание количества мегабайтов дисковой памяти, отводимой под "несжатый" диск. Для того, чтобы отвести под создаваемый "сжатый" диск как можно больше свободного пространства, указывайте параметр *размер* равным 0. По умолчанию параметр *размер* принимается равным 1 MB. Допустимо использование вместо ключа /RESERVE ключа /RE;

/SIZE = *размер* — указание в мегабайтах полного размера создаваемого "сжатого" диска. По умолчанию этот *размер* равен 1 MB. Вместо ключа /SIZE можно использовать ключ /SI.

Примеры использования:

Для создания нового "сжатого" диска с использованием всей свободной поверхности на диске E: введите следующую команду:

```
dblspace /CREATE e: /RESERVE = 0
```

Для проведения приведенной выше операции, но с установкой для диска E: объема в 12 MB введите команду:

```
dblspace /CREATE e: /SIZE = 12
```

Если же вам необходимо, например, зарезервировать для "несжатого" диска допустим 2.75 MB дискового пространства, можно ввести команду:

```
dblspace /CREATE d: /RESERVE = 2.75
```

После выполнения этой команды будет создан "сжатый" диск D: и на "горячем" диске C будет зарезервировано 2.75 MB дискового пространства.

Для создания нового "сжатого" диска D: объемом 1 MB просто введите:

```
dblspace /CREATE D:
```

DBLSPACE/DEFRAGMENT

Проведение операции дефрагментации выбранного "сжатого" диска.

Формат команды:

dblspace /DEFRAGMENT *диск:*

Параметры:

диск: - выбор диска, подлежащего дефрагментации. По умолчанию для дефрагметации выбирается текущий диск.

Ключ

/DEFRAGMENT	проведение операции дефрагментации диска. Допускается использование вместо ключа /DEFRAGMENT ключа /DEF.

Примечание:

- Различия между утилитой DEFRAG и командой DBLSPACE/DEFRAGMENT. При использовании утилиты DEFRAG она производит физическую реорганизацию файлов на диске с целью увеличения скорости доступа к файлам, не затрагивая "сжатый" диск, который в общем случае представляет собой просто файл. Однако, когда вы используете команду DBLSPACE/DEFRAGMENT, то в процессе работы этой команды производится дефрагментация файлов внутри именно "сжатого" файла.

Пример использования:

dblspace /DEFRAGMENT d:

После ввода этой команды Double Space организует процесс дефрагментации файлов внутри "сжатого" диска D:.

DBLSPACE/DELETE

Удаление созданного "сжатого" диска и уничтожение файла "сжатого" тома - (CVF). Уничтожение сжатого диска уничтожает также и файлы, которые находятся на нем.

Формат команды:

dblspace /DELETE *диск:*

Параметр:

диск: - указание буквы диска, подлежащего удалению. Нельзя удалить диск C:.

Ключи

/DELETE	удаление выбранного "сжатого" диска, выбранного в качестве параметра команды. Вместо /DELETE можно использовать ключ /DEL.

Пример использования:

dblspace /DELETE d:

Производится удаление "сжатого" диска D:.

Примечания:

Если вы в процессе работы удалили "сжатый" диск, вы можете использовать для его восстановления утилиту Undelete. Файл, содержащий в себе данные "сжатого" диска, имеет имя, записанное в следующем формате:

dblspace.xxx, (например dblspace.000)

После восстановления этого файла вам необходимо запустить утилиту DBLSPACE в режиме MOUNT командой

dblspace /MOUNT диск:

Подробности работы Double Space в режиме MOUNT описаны далее.

DBLSPACE/FORMAT

Форматирование выбранного "сжатого" диска.

Предупреждение:

В процессе форматирования все данные, находящиеся на выбранном диске, будут уничтожены.

Формат команды:

dblspace /FORMAT *диск:*

Параметр:

диск: - указание буквы логического "сжатого" диска, над которым будет выполняться операция форматирования.

Ключ

/FORMAT	форматирование выбранного "сжатого" диска. Допустимо использовать вместо ключа /FORMAT ключ /F

Пример использования:

dblspace /FORMAT e:

Производится форматирование "сжатого" диска E:.

Примечание:

Нельзя форматировать "сжатый" диск C:.

DBLSPACE/INFO

Получение информации о выбранном "сжатом" диске.

Формат команды:

dblspace /INFO *диск:*

Параметр:

диск: - указание буквы, выбираемого сжатого диска.

Ключ

/INFO получение информации о выбранном в качестве параметра диске. По умолчанию ключ /INFO установлен для любого "сжатого" диска при вводе команды DBLSPACE. Если параметр "диск:" не указан, то информация выводится о текущем диске.

Пример использования:

dblspace /INFO c:

Получение информации о "сжатом" диске C:. Для выполнения этой же операции можно просто ввести:

dblspace c:

DBLSPACE/LIST

Вывод на экран всех имеющихся на вашем компьютере логических дисков, за исключением сетевых.

Формат команды:

dblspace /LIST

Ключи

/LIST вывод на дисплей всех логических дисков системы, за исключением сетевых, а также всех других доступных системе дисков, отличных по своей конструкции от жестких или гибких дисков.

Пример использования:

dblspace /LIST

После выполнения этой команды ОС выведет на дисплей список всех доступных системе логических дисков.

DBLSPACE/MOUNT

Установка соединения между файлом "сжатого" тома (CVF) и идентификатором логического диска, на котором расположен файл CVF. Double Space обычно сам автоматически устанавливает соответствие между CVF и логическим диском.

Операция соединения CVF и логического диска бывает необходима в тех случаях, если вы "отсоединили" CVF от логического диска или если CVF расположен на гибком диске.

Формат команды:

dblspace /MOUNT = *ппп диск1:* /NEWDRIVE = *диск2:*

Параметры:

диск1: - указание буквы логического диска, который должен быть соединен с CVF.

Ключи

/MOUNT	указание расширения CVF. К примеру, если вы хотите соединить с логическим диском,файл CVF с именем DBLSPACE.001, то вы должны указать ключ /MOUNT = 001. Если вы опускаете этот параметр, то Double Space пытается присоединить к логическому диску файл с именем DBLSPACE.000. Допускается использование вместо ключа /MOUNT ключа /MO;
/NEWDRIVE = *диск2:*	указание буквы алфавита, которой должен обозначаться логический диск, присоединяемый к "сжатому" диску. По умолчанию логическому диску, присоединяемому к "сжатому" диску, присваивается буква алфавита, следующая по порядку за буквой, обозначающей последний диск в системе. Допускается вместо ключа /NEWDRIVE использовать аббревиатуру /NEW.

Примеры использования:

Для присоединения CVF к логическому диску A: введите следующую команду:

```
dblspace /MOUNT a:
```

Если вам необходимо присоединить к логическому диску D: CVF с именем DBLSPACE.001, необходимо ввести команду:

```
dblspace /MOUNT = 001 d:
```

DBLSPACE /UNMOUNT

Формат команды:

dblspace /UNMOUNT *диск:*

Прерывание соединения между выбранным "сжатым" диском и логическим диском.

Параметр:

диск: - выбранный для отсоединения диск. Если параметр *диск:* не указан, то по умолчанию принимается текущий диск.

Ключ

/UNMOUNT	отсоединение выбранного в параметре *диск:* "сжатого" диска. Допускается по умолчанию ключ /U.

Пример использования:

dblspace /UNMOUNT a:

После выполнения этой команды от логического диска A: будет отсоединен CVF.

Примечание:

Нельзя отсоединить с помощью команды DBLSPACE/UNMOUNT логический диск C:.

DBLSPACE/RATIO

Выбор коэффициента сжатия данных на выбранном "сжатом" диске.

Формат команды:

dblspace /RATIO = *r.r диск:* /ALL

Параметры:

диск: - указание буквы выбираемого логического диска. В этом параметре можно указывать несколько логических дисков, перечислив их обозначения через пробел. Если вам необходимо изменить коэффициент сжатия данных на всех "сжатых" дисках, то вам необходимо использовать ключ /ALL. Если же параметр *диск:* опущен, то Double Space будет изменять коэффициент сжатия только на текущем диске.

Ключи

/RATIO = r.r	указание коэффициента сжатия. Коэффициент сжатия изменяется в пределах от 1.0 до 16.0. Если вы не используете этот ключ, то Double Space сам установит оптимальный коэффициент сжатия данных;

Примеры использования:

dblspace /RATIO d:

После выполнения этой команды DBLSPACE автоматически установит оптимальный коэффициент сжатия данных для диска D:. Если необходимо изменить коэффициент сжатия данных на диске D: на 3.2, введите следующую команду:

dblspace /RATIO = 3.2 d:

Для изменения коэффициента сжатия для текущего диска на 6 введите такую команду:

dblspace /RATIO = 6.0

DBLSPACE/SIZE

Установка или изменение размера "сжатого" диска. Если логический диск, к которому присоединен CVF, имеет дополнительное свободное место, то можно использовать свободное дисковое пространство.

dblspace /SIZE = *размер1* /RESERVE = *размер2 диск:*

Параметр:

диск: - указание буквы, обозначающей выбранный для изменения размера "сжатый" диск.

Ключи

/SIZE = *размер1*	указание нового размера выбранного в качестве параметра "сжатого" диска. Допускается использование вместо ключа /SIZE ключа /SI;
/RESERVE = *размер2*	указание размера дисковой поверхности в мегабайтах, резервируемой под "несжатый" диск. Допускается использование вместо ключа /RESERVE ключа /RES.

Примеры использования:

Для того, чтобы изменить размер "сжатого" диска C:, например, и для резервирования 60 MB дискового пространства под "несжатый" диск D:, необходимо ввести следующую команду:

dblspace /SIZE = 40 /RESRVE = 60 d:

Если же вам необходимо использовать под "сжатый" диск все свободное дисковое пространство диска D:, введите команду:

dblspace /SIZE /RESERVE = 0 c:

DEBUG

Запуск отладчика DEBUG, который можно использовать для отладки и проверки выполняемых программ. После запуска DEBUG устанавливает свою собственную среду, в которой пользователь может с помощью команд отладчика управлять проверкой и отладкой выполняемых программ. После запроса DEBUG вы можете вводить его команды с параметрами.

Формат команды:

debug *диск:\маршрут\имя_тестир_файла параметры_тестир_файла*

Параметры:

диск:\маршрут\имя_тестир_файла - указание полного идентификатора проверяемого выполняемого файла.

параметры_тестир_файла - указание параметров командной строки для выполняемого файла.

Команды отладчика DEBUG			
A	(Assemble)	P	(Proceed)
C	(Compare)	Q	(Quit)
D	(Dump)	R	(Register)
E	(Enter)	S	(Search)
F	(Fill)	T	(Trace)
H	(Hex)	U	(Unassemble)
G	(Go)	W	(Write)
I	(Input)	XA	(Allocation Expanded Memory)
L	(Load)	XD	(Reallocation Expanded Memory)
M	(Move)	XM	(Map Extended Memory Pages)
N	(Name)	XS	(Display Expanded Memory Status)
O	(Output)		

DEBUG: A (Assemble)

Запись мнемонических команд языка Ассемблер для процессоров 8086, 8087, 8088 в оперативную память. Эта команда позволяет с помощью мнемоники Ассемблера и его структур создать выполняемые машинные коды. Все числовые данные должны заносится в шестнадцатерич-

ном формате. Для этого должно использоваться от 1 до 4 цифр. Каждая команда Ассемблера должна содержать обязательно какую-либо машинную команду в символьном виде.

Формат команды:

A *адрес*

Параметры:

адрес - указание адреса оперативной памяти, с которого вы будете располагать в памяти создаваемую программу. Для указания адреса памяти необходимо использовать числа в шестнадцатеричном формате, оканчивая строку символов, содержащую адрес символом "h". Если вы не укажете в команде A параметр *адрес*, то команда DEBUG A выполняться не будет.

Примеры использования:

Команда DEBUG A поддерживает все типы адресации данных, что иллюстрирует нижеприведенный пример:

```
ADD     BX,34[BP+2].[SI-1]
POP     [BP+DI]
PUSH    [SI]
```

Команда A также поддерживает и все другие мнемокоды Ассемблера:

```
LOOPZ   100
LOOPE   100
JA      200
JNBE    200
```

Для организации работы с математическим сопроцессором с помощью мнемокодов Ассемблера используйте префиксы WAIT и FWAIT, как в следующем примере:

```
FWAIT FADD ST,ST(S); this line assemble an fwait prefix
```

Примечание к этой команде гласит, что в этом мнемокоде Ассемблера используется префикс FWAIT для работы с математическим сопроцессором.

Примечания:

Использование мнемоники. При использовании мнемоники Ассемблера в команде DEBUG A допускается не использовать обозначения регистров CS:, DS:, ES:, SS:.

Для возврата из дальних процедур необходимо использовать процедуру RETF. Для работы со строками вы должны ясно указывать размер данных. В качестве примера можно привести команду MOVSW для

пересылки машинного слова (16 бит) и команду MOVSB для пересылки одного байта (8 бит).

Программирование переходов и вызовов процедур. Ассемблер автоматически определяет тип вызова процедуры или перехода: короткий, ближний или дальний, в зависимости от количества байтов перехода на конечный адрес. Однако при организации перехода или вызова процедуры вы можете явно указывать тип перехода или процедуры при помощи префиксов NEAR или FAR, как это сделано в следующем примере:

```
-a:0100:0500
0100:0500     JMP    502              ; a 2 - byte short jump
0100:0502     JMP    NEAR 505         ; a 3 - byte near jump
0100:0505     JMP    FAR 50A          ; a 5 - byte far jump
```

Для упрощения ввода данных вместо аббревиатуры NEAR возможно использовать префикс NE.

Когда вы хотите записать слово или байт в память, вы должны указать префикс, определяющий тип данных, WORD PTR или BYTE PTR. Допускается использование для приведенных ранее префиксов аббревиатур WO или BY соответственно. Пример, приведенный ниже, показывает 2 типа форматов определения данных:

```
DEC WO [SI]
NEG BYTE PTR [128]
```

Отладчик DEBUG использует для указания операндов общепринятый формат, когда операнды заключаются в скобки, но вы можете указывать операнды и не указывая скобок " [" и "]".

Для примера ниже приводятся два формата задания данных:

```
MOV     AX,[21]
MOV     AX,21
```

Для работы со строками в команде A отладчика DEBUG вам доступны 2 популярные инструкции:

- DB, которая позволяет разместить данные в памяти побайтно;

- DW, которая размещает данные в памяти машинными словами.

В качестве примера приводятся следующие инструкции:

```
DB 1,2,3,4, " This is an example "
DB ' This is a quatation mark: "
DB @ This is a quatation mark:'"
DW 1000,2000,3000, "BACH"
```

DEBUG C (Compare)

Сравнение данных в двух областях памяти.

Формат команды:

C *диапазон адрес:*

Параметры:

диапазон - указание стартового и конечного адреса области оперативной памяти или стартового адреса и длины области памяти, если первая область памяти выбрана для сравнения.

адрес: - указание стартового адреса второй области памяти, выбранной для сравнения.

Примеры использования:

Следующие команды выполняют одну и ту же операцию:

```
C 100, 10F 300
C 100l10 300
```

Каждая из приведенных команд сравнивает блок памяти, начиная с адреса 100h по адрес 10Fh и блок памяти начиная с адреса 300 по 30Fh. После ввода одной из этих команд на дисплей будет выведена следующая информация:

```
197F:0100  4D    E4  197F:0300
197F:0101  67    99  197F:0301
197F:0102  A3    27  197F:0302
197F:0103  35    F3  197F:0303
197F:0104  97    BD  197F:0304
197F:0105  04    35  197F:0305
197F:0107  76    71  197F:0307
197F:0108  E6    11  197F:0308
197F:0109  19    2C  197F:0309
197F:010A  80    0A  197F:030A
197F:010B  36    7F  197F:030B
197F:010C  BE    22  197F:030C
197F:010D  83    93  197F:030D
197F:010E  49    77  197F:030E
197F:010F  4F    8A  197F:030F
```

Значение 197F в отчете команды DEBUG C соответствует записанному в регистре DS значению 197F.

Заметьте, что в приведенном отчете отсутствуют данные, находящиеся в ячейках памяти с адресами 197F:0106 и 197F:0306. Это означает, что в ячейках памяти с указанными адресами данные совпадают.

DEBUG :D (Dump)

Вывод на дисплей дампа (содержимого) оперативной памяти.

Формат команды:

D *диапазон*

Параметр:

диапазон - указание стартового и конечного адресов области памяти, или стартового адреса и длины области памяти, с которой необходимо получить ее дамп.

Если в команде параметр "диапазон" опущен, DEBUG выводит на экран дисплея содержимое ближайших старших 128 байт памяти, начиная отчет с конечного адреса, заданного предыдущей командой DEBUG D.

Примеры использования:

Введите команду:

```
Dcs:100 10AF
```

В процессе ее выполнения DEBUG выведет на экран следующую информацию в следующем формате:

```
04BA:0100 54 4F 4D 00 53 41 57 59-45 52 00 00 00 00 00 00

TOM.SAWYER.....
```

Каждая линия отчета команды DEBUG D начинается с указания адреса памяти, а затем указания содержимого следующих 16 байт данных или 8 байт данных при использовании режима дисплея 40×25 символов.

Для вывода 20h байтов начиная с адреса 100h по 115h в сегменте CS (сегменте команд) введите команду:

```
Dcs:100 115
```

Примечание:

Когда вы используете команду DEBUG D, отладчик выводит информацию в двух форматах: в виде шестнадцатеричных чисел и в символах ASCII. Это связано с тем, что каждому шестнадцатеричному коду DEBUG ставит в соответствие символ ASCII с тем же кодом.

DEBUG: E (Enter)

Ввод данных в память, начиная с указанного адреса памяти. Вы можете указывать вводимые данные в шестнадцатеричном формате или символами ASCII.

Формат команды:

E *адрес данные*

Параметры:

адрес - указание начального адреса оперативной памяти, куда будет осуществляться ввод данных.

данные - указание данных, которые необходимо ввести в память.

Примеры использования:

Для примера введите команду:

```
Ecs:100
```

DEBUG выведет на дисплей сообщение в следующем формате:

```
04BA:0100 EB._
```

для изменения текущего значения ячейки памяти с адресом 04BA:0100 на 41, наберите 41 после запроса DEBUG и нажмите клавишу Enter:

```
04BA:0100 EB.41_
```

Вы можете изменять значения и других ячеек памяти в одной команде E. После нажатия клавиши Enter для ввода нового значения в предыдущем примере нажмите для примера 3 раза клавишу SpaceBar. Debug выведет на дисплей следующие значения:

```
04BA:0100 EB.41 10. 00. BC._
```

Для изменения шестнадцатеричного значения BC на 42, наберите 42 после курсора и нажмите Enter:

```
04BA:0100 EB.41 10. 00. BC.42_
```

Если вам необходимо изменить значение 10 на 6F в ячейке памяти с адресом 0101 нажмите клавишу "стрелка вниз" и удерживайте до тех пор, пока DEBUG не выведет содержимое ячейки памяти с адресом 0101.

Измените значение 10h на 6Fh, как описана эта операция раньше. DEBUG выведет на дисплей сообщения в следующем формате:

```
04BA:0100 EB.41 10. 00. BC.42-
04BA:0102 00.-
04BA:0101 10._
```

Наберите новое значение 6F:

```
04BA:0101 10.6F_
```

Далее нажмите клавишу Enter для окончания работы команды E и возврата в среде DEBUG.

Для ввода строковых данных пользуйтесь командами в нижеприведенном формате:

```
Eds:100 "This is the text example"
```

Эта команда помещает 24 байта в оперативную память, начиная с адреса DS:100.

Примечания:

- Если вы указываете в команде E параметр *адрес* без указания данных, подлежащих вводу, DEBUG начинает выводить на дисплей содержимое ячейки памяти, начиная с текущего адреса, и будет ожидать ввода вами информации.
- Если вы указываете значения в параметре *данные*, команда E автоматически заносит эти данные в оперативную память, начиная с указанного адреса. Если при вводе данных будет обнаружена ошибка, то данные не будут изменены.

DEBUG: F (Fill)

Заполнение специальных областей памяти значениями пользователя DEBUG. При использовании этой команды разрешается вводить данные в шестнадцатеричном формате или с помощью символов ASCII.

Формат команды:

F *диапазон значения*

Параметры:

диапазон - указание стартового и конечного адресов области памяти или стартового адреса и длины области специальной памяти, которую необходимо заполнить значениями пользователя.

значения - указание данных, подлежащих вводу. Этот параметр может включать в себя данные, как в шестнадцатеричном формате, так и строковые данные, заключенные в кавычки.

Пример использования:

```
COMD3 = F04BA:100l100 42 45 52 54 41
```

В приведенном примере DEBUG добавляет в специальные адреса памяти, начиная с 04BA:100 по 04BA:1FF пять специальных значений.

Примечания:

- Если указанный диапазон памяти не существует, или в него нельзя ввести данные, то DEBUG выведет на дисплей сообщение об ошибке, и работа команды DEBUG F будет прервана.
- Если вы включили в команду больше байтов, чем их может вместить указанный диапазон адресов памяти, DEBUG игнорирует значения, указанные сверх нормы.

DEBUG: G (Go)

Запуск выполняемых программ.

Формат команды:

G = *адрес точки_прерыв*

Параметры:

адрес - указание адреса выполняемой программы в памяти, с которого должно начинаться ее выполнение. Если вы не указываете этого адреса, DEBUG попытается выполнить программу с текущего адреса, установленного в регистрах CS:IP.

точки_прерыв - задание от 1 до 10 точек прерывания, которые могут быть использованы при работе с командой G.

Примеры использования:

Для запуска программы с текущего адреса 7500, установленного в сегменте CS, введите следующую команду:

```
Gcs:7500
```

После запуска программы DEBUG будет выводить текущее состояние всех регистров и регистра флагов.

Для задания точки прерывания введите команду:

```
Gcs:7500 cs:7550,cs:8000
```

В этой команде задаются адреса двух прерываний - cs:7550, cs:8000.

DEBUG: H (Hex)

Выполнение команд сложения и вычитания над двумя шестнадцатеричными числами.

Формат команды:

H *значение1 значение2*

Параметры:

значение1 - указание первого шестнадцатеричного числа в диапазоне от 0 до FFFFh.

значение2 - указание второго шестнадцатеричного числа в диапазоне от 0 до FFFFh.

Примеры использования:

H19F 10C

После ввода этой команды DEBUG вычисляет и выводит на дисплей следующие результаты:

02A9 0095

Примечания:

При работе этой команды DEBUG сначала производит операцию суммирования двух указанных чисел и вычитает из первого числа второе. Результаты вычислений отображаются на дисплее в одну строку: сумма записана первой, разность является вторым значением.

DEBUG: I (Input)

Чтение и вывод на дисплей одного байта информации из указанного порта.

Формат команды:

I *порт*

Параметр:

порт - указание адреса входного порта, из которого будет осуществляться чтение и вывод на дисплей 1 байта.

Примеры использования:

Допустим, вам необходимо прочитать 1 байт из порта с адресом 2F8h. Введите команду:

I2F8

По этой команде DEBUG читает 1 байт информации из порта с указанным адресом и выводит его на дисплей. Если в указанном порту записано, например, число 42h, то на дисплее появится следующее сообщение:

42

DEBUG: L (Load)

Загрузка файла или нескольких указанных файлов на диске в память. Для загрузки файла или секторов используются адреса, заданные в регистрах BX:CX для конкретного дискового файла.

Формат команды:

L *адрес*

Для того чтобы не использовать файловую систему MS-DOS для загрузки указанных секторов, используйте следующий формат:

L *адрес диск старт номер*

Параметры:

адрес - указание адреса памяти, с которого будет производиться загрузка в память файла или указанных секторов. Если этот параметр не указан, то DEBUG начнет загрузку файла (секторов) в память с текущего адреса, установленного в регистре CS.

диск - указание логического диска, с которого необходимо считать указанное число секторов. Этот параметр задается в следующем формате: 0=A, 1=B, 2=C и т.д. Для загрузки секторов с указанного диска с конкретного файла вы можете задавать эти параметры в командной строке или использовать для этой цели команду DEBUG N (Name).

старт - указание номера первого сектора, подлежащего загрузке в память в шестнадцатеричном виде.

номер - указание числа секторов, которые необходимо загрузить в память, включая первый сектор, указанный в параметре *старт*.

Примеры использования:

Для загрузки в память файла FILE.COM с диска введите следующую команду:

```
Lfile.com
```

В процессе выполнения этой команды DEBUG загрузит FILE.COM в память с текущего адреса относительно регистра CS, и вернется к своему запросу. Если вам, например, необходимо загрузить 109 секторов с диска C, начиная с логического сектора 15 (0Fh), и расположить их в памяти, начиная с адреса 04BA:0100, введите команду:

```
L04BA:0100 2 0F 6D
```

Примечания:

- Когда вы используете команду L без параметров, DEBUG загружает указанный файл в память, начиная с адреса CS:100. DEBUG также устанавливает в регистрах BX и CX значения, соответствующие

числу загруженных в память байтов. Если в команде L вы не указываете имя файла, то тогда загружается файл, указанный в последней команде DEBUG:N.

- Если в команде L вы не указываете параметр *адрес*, DEBUG загружает файл или указанное число секторов в адрес памяти, указанный в регистре CS.
- При загрузке файлов с расширением .EXE DEBUG игнорирует адрес, с которого файл должен загружаться в память. При загрузке .EXE файла DEBUG сам производит настройку адресов памяти согласно заголовку .EXE файла. Если вы далее хотите проверить правильность организации .EXE файла, переименуйте его и загрузите его с другим расширением.
- Стандартный шестнадцатеричный файл использует стандартный шестнадцатеричный формат фирмы Intel, подробно описанный в книге "Энциклопедия MS-DOS". DEBUG считает, что файлы с расширением .HEX записаны в шестнадцатеричном формате. Команда L может работать с шестнадцатеричными файлами. При работе с такими файлами адрес загрузки можно не указывать, так как этот адрес должен быть указан в загружаемом файле. Если же параметр *адрес* используется для загрузки шестнадцатеричных файлов, то в этом случае DEBUG добавляет указанный адрес к адресу, указанному в загружаемом файле.

DEBUG: M (Move)

Копирование указанного блока памяти в другой блок памяти.

Формат команды:

M *диапазон адрес*

Параметры:

диапазон - указание стартового и конечного адресов области памяти или стартового адреса и длины области памяти, подлежащей копированию.

адрес - указание стартового адреса памяти, куда необходимо скопировать указанные в параметре *диапазон* области памяти.

Примеры использования:

Допустим, вам необходимо скопировать блок памяти, начиная с адреса CS:110 в адрес CS:510 и далее скопировать все байты за CS:100 в старших адресах памяти до адреса CS:10F. Скопированные данные должны располагаться в памяти, начиная с адреса CS:510 по адрес CS:51F. Используйте для решения этой задачи команду:

```
M:100 10F 510
```

Для просмотра результатов копирования данных используйте команду DEBUG D (Dump), указывая адреса просмотра областей памяти, использованные в команде DEBUG M.

Примечания:

Если данные указанного блока памяти скопированы не до конца, то оригинальные данные в области памяти, куда копируются байты информации, будут оставаться нетронутыми. Если в блоке памяти, предназначенном для размещения копируемых данных, имеются какие-либо данные, то новые данные будут записаны поверх старых.

DEBUG: N (Name)

Указание имени выполняемого файла для использования в дальнейшем командами DEBUG L (Load) или W (Write) или задание параметров выполняемого файла, который будет отлаживаться.

Формат команды:

N *диск:\маршрут\имя_файла*

Для указания в этой команде параметров выполняемого файла используйте следующий формат:

N *параметры_файла*

Параметры:

диск:\маршрут\имя_файла - указание полного идентификатора тестируемого файла.

параметры_файла - указание параметров командной строки выполняемого файла, которые передаются ему при загрузке.

Примеры использования:

Для загрузки программы PROG.COM для отладки и задания после этого ей параметров используйте следующую последовательность команд:

```
DEBUG PROG.COM
N параметр1 параметр2
```

В этом примере команда G запускает программу PROG.COM на выполнение, а команда N передает ей параметры: *параметр1* и *параметр2*. В результате выполнения вышеперечисленных команд отладчика DEBUG запуск программы PROG.COM будет аналогичен ее запуску из командной строки DOS:

```
PROG.COM параметр1 параметр2
```

Примечания:

- Команда N отладчика DEBUG может использоваться в двух вариантах. В первом варианте командой N задается имя файла для дальнейшего использования его командой L (Load) или W (Write). Во втором случае использования команды N она передает загружаемому файлу параметры командной строки.
- Для загружаемого файла эта команда использует специальные адреса, значения байтов в которых указаны ниже:

  CS:5C - блок управления первым файлом FCB1;

  CS:6C - блок управления вторым файлом FCB2;

  CS:80 - длина в символах командной строки (число n);

  CS:81 - начало командной строки для первого файла.

DEBUG: O (Output)

Запись байта в указанный выходной порт.

Формат команды:

O *порт байт*

Параметры:

порт - указание адреса порта, куда необходимо занести байт информации. Адрес выходного порта указывается в шестнадцатеричном виде.

байт - указание байта информации, подлежащей записи в порт.

Пример использования:

Для записи значения 4F в выходной порт с адресом 2F8 наберите команду:

```
O2F8 4F
```

DEBUG: P (Proceed)

Организация выполнения циклов, повторяющихся команд, программных прерываний или подчиненных процедур, контроль выполнения других команд.

Формат команды:

P = *адрес номер*

Параметры:

адрес - указание адреса первой выполняемой команды. Если адрес не указан, то по умолчанию начальным адресом в этой команде принимается текущий адрес, установленный в регистрах CS:IP.

номер - указание числа команд, которые необходимо выполнить под управлением команды P отладчика DEBUG. По умолчанию этот параметр равен 1. Для получения информации о запуске программ в памяти посмотрите описание команды DEBUG G (Go).

Примеры использования:

Допустим, вам необходимо выполнить в проверяемой программе инструкцию CALL по адресу CS:143F. Для запуска подчиненной процедуры с именем CALL и возвращения управления DEBUG после ее выполнения введите следующую команду:

```
P = 143F
```

DEBUG выведет результаты выполнения этой команды в виде:

```
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000
DI=0000
DS=2246 ES=2246 SS=2246 CS=2246 IP=1443 NV UP EI PL NZ AC
PO NC
2246:1442 7505 JNZ 144A
```

Примечания:

- Когда в команде P отладчик DEBUG организует проверку программы, он не прерывает ее до тех пор, пока не встретит команды организации циклов, повторяющиеся команды, программные прерывания или подчиненную процедуру, адрес которой совпадает с указанным, или пока указанный машинный код не совпадет с указанным в команде P.
- Если параметр *адрес* в сегменте CS не указан, то DEBUG использует текущее значение, установленное в регистрах CS:IP. Если указанные в команде P адреса не являются адресами команд цикла, повторяющихся команд, программных прерываний или подчиненных процедур, то команда P работает как команда DEBUG:T (Trace).
- После выполнения команды P она выводит на дисплей содержимое программных регистров, состояние регистра флагов и декодированную форму следующей выполняемой команды.
- Вы не можете использовать команду P для трассировки (выполнения по шагам) команд системного ПЗУ (ROM BIOS).

DEBUG: Q (Quit)

Выход из программы DEBUG в среду MS-DOS.

Формат команды:

```
Q
```

Пример использования:

Для окончания работы с DEBUG введите команду:

Q

MS-DOS после выполнения этой команды выведет свой системный запрос.

DEBUG: R (Registers)

Вывод на дисплей содержимого одного или нескольких регистров центрального процессора (Central Prossesing Unit - CPU).

Формат команды:

R *имя_регистра*

Для вывода на дисплей содержимого всех программных регистров и регистра флагов используйте следующий формат команды:

R

Параметры:

имя_регистра - указание условного обозначения регистра, содержимое которого необходимо вывести на дисплей.

Пример использования:

Допустим, необходимо просмотреть содержимое программных регистров и регистра флагов, а также вид следующей команды в мнемонических кодах Ассемблера. Для решения этой задачи введите команду:

R

Если текущим адресом например является CS:011A, то на дисплей будет выведено следующее сообщение:

```
AX=0E00 BX=00FF CX=0007 DX=01FF SP=039D BP=0000 SI=005C
DI=0000
DS=04BA ES=04BA SS=04BA CS=04BA IP=011A NV UP DI NG NZ
AC PE NC
04BA:011A CD21 INT 21
```

Для просмотра статуса регистра флагов введите команду:

Rf

DEBUG выведет сообщение в следующем формате:

```
NV UP DI NG NZ AC PE NC -_
```

Если вы хотите изменить содержимое одного или нескольких байтов в регистре флагов, введите команду:

```
NV UP DI NG NZ AC PE NC - pleycy
```

DEBUG выполнит эту операцию и приостановит выполнение команды R, а затем выведет свой запрос. Для просмотра изменений введите команду R или Rf. Сообщение DEBUG в этом случае будет иметь вид:

```
NV UP EI PL NZ AC PE CY -
```

Далее нажмите клавишу Enter для возвращения в среду DEBUG.

DEBUG: S (String)

Организация поиска в заданном диапазоне адресов памяти одного или нескольких указанных байтов.

Формат команды:

S *диапазон данные*

Параметры:

диапазон - указание начального и конечного адресов области памяти для организации поиска.

данные - указание байта (байтов) информации или строки символов, которые надо найти. Строка символов должна быть заключена в кавычки.

Примеры использования:

Допустим, вам необходимо найти значение 41 во всем диапазоне адресов от CS:100 до CS:110. Наберите команду:

```
SCS:100 110 41
```

После выполнения этой команды DEBUG выведет отчет о своей работе в следующем формате:

```
04BA:0104
04BA:01D
```

Если же у вас возникла необходимость найти строку "Ph" в том же диапазоне, что и в ранее приведенном примере, введите команду:

```
SCS:100 110 "Ph"
```

Примечание:

В выводимом на дисплей списке адресов DEBUG выводит на дисплей те адреса памяти, в которых найдено указанное в команде S значение.

DEBUG: T (Trace)

Выполнение программ в режиме трассировки. В этом режиме работы DEBUG выполняет одну машинную команду и далее осуществляет вывод на дисплей содержимого всех программных регистров, статуса регистра флагов, а также мнемонический код Ассемблера для следующей выполняемой команды.

Формат команды:

T = *адрес номер*

Параметры:

адрес - указание адреса первой выполняемой команды. Если адрес не указан, то по умолчанию начальным адресом в этой команде принимается текущий адрес, установленный в регистрах CS:IP.

номер - указание числа команд, которые необходимо выполнить под управлением команды T отладчика DEBUG. По умолчанию этот параметр равен 1. Для получения информации о запуске программ в памяти посмотрите описание команды DEBUG G (Go).

Пример использования:

Для выполнения одной машинной команды и просмотра состояния всех программных регистров, статуса регистра флагов и следующей декодированной команды введите:

```
T
```

В этом примере выполняется команда, расположенная по текущему адресу, указанному в CS:IP. Допустим, что он равен 04BA:011A. Тогда отчет команды DEBUG T будет иметь следующий вид:

```
AX=0E00 BX=00FF CX=0007 DX=01FF SP=039D BP=0000 SI=005C
DI=0000
DS=04BA ES=04BA SS=04BA CS=04BA IP=011A NV UP DI NG NZ
AC PE NC
04BA:011A CD21 INT 21
```

Примечания:

- Команда T отладчика DEBUG использует для своей работы режим пошагового выполнения команд микропроцессоров 8086 и 8088.
- С помощью этой команды вы имеете возможность прочитать содержимое системного ПЗУ.

DEBUG: U (Unassemble)

Дизассемблирование машинных кодов и вывод на дисплей команд Ассемблера, его директив для организации структуры программы вместе с адресами команд и значениями операндов.

Формат команды:

U *диапазон*

По умолчанию, если параметр *диапазон* дизассемблируется 20h байтов, начиная с первого адреса, заданного предыдущей командой U. Для использования этого варианта формат команды DEBUG U имеет вид:

```
U
```

Параметр:

диапазон - указание стартового и конечного адресов области памяти или стартового адреса и длины области памяти, для которой необходимо выполнить дизассемблирование.

Пример использования:

Для дизассемблирования 16 (10h) байтов, начиная с адреса 04BA:0100, введите команду:

```
U04BA:0100l10
```

DEBUG выведет отчет о своей работе в следующем формате:

```
04BA:0100 206472 AND [SI+72],AH
04BA:0103 69 DB 69
04BA:0104 7665 JBE 016B
04BA:0106 207370 AND [BP+DI+70],DH
04BA:0109 65 DB 65
04BA:010A 63 DB 63
04BA:010B 69 DB 69
04BA:010C 66 DB 66
04BA:010D 69 DB 69
04BA:010E 63 DB 63
04BA:010F 61 DB 61
```

Для вывода на дисплей дизассемблированных машинных кодов в области памяти, начиная с адреса 04BA:0200 по 04BA:0208, введите такую команду:

```
04BA:0100 206472 AND [SI+72],AH
04BA:0103 69 DB 69
04BA:0104 7665 JBE 016B
04BA:0106 207370 AND [BP+DI+70],DH
```

DEBUG: W (Write)

Запись файла или указанного числа секторов из оперативной памяти на диск. Вы должны указать имя дискового файла либо в самой команде W, либо с помощью команды DEBUG N. Команда использует для организации работы с файлом метод FCB. Сам блок управления файлом будет расположен по адресу CS:5C. Количество байтов, записываемых на дисковый файл, указано в регистрах BX:CX.

Формат команды:

W *адрес*

Для того, чтобы не использовать файловую систему MS-DOS при записи определенного числа секторов на диск, используйте команду W в следующем формате:

W *адрес диск старт номер*

Параметры:

адрес - указание адреса памяти, с которого начинается файл или его часть, подлежащая записи в дисковый файл. Если адрес не указан, DEBUG считает начальным адресом - адрес CS:100.

диск - указание логического диска, с которого необходимо считать указанное число секторов. Этот параметр задается в следующем формате: 0=A, 1=B, 2=C и т.д.

старт - указание номера первого сектора, подлежащего загрузке в память в шестнадцатеричном виде.

номер - указание числа секторов, которые необходимо загрузить в память, включая первый сектор, указанный в параметре *старт*.

Примеры использования:

Допустим, вам необходимо записать в дисковый файл данные, начиная с адреса CS:0200, на логический диск B. Данные должны располагаться на диске, начиная с сектора 37h, и продолжаться 2Bh секторов. Для выполнения этой операции введите команду:

```
WCS:100 1 37 2B
```

После выполнения операции DEBUG выведет на дисплей свой запрос.

Примечания:

- Если вы использовали команды DEBUG G, T, P или R, вам необходимо обнулить регистр BX:CX перед использованием команды W без параметров.

- Если вы модифицировали файл, не изменяя его имени, длины и стартового адреса, DEBUG сам корректно записывает этот файл в то же оригинальное место на диске.
- Команда DEBUG не позволяет записывать файлы в формате .EXE и .HEX.

DEBUG: XA (Allocation Expanded Memory)

Указание номера страницы расширенной памяти для работы с DEBUG. Для использования расширенной памяти вы должны инсталлировать драйвер расширенной памяти перед использованием отладчика DEBUG. Этот драйвер должен удовлетворять стандарту LIM EMS 4.0.

Формат команды:

XA *номер*

номер - указание номера страницы 16-килобайтной страницы EMS, которая будет использоваться для работы.

Примеры использования:

Для обращения к 8-й странице EMS введите команду:

```
XA8
```

Если команда выполнится успешно, DEBUG выводит на дисплей следующее сообщение:

```
Handles created = 0003
(канал создан с номером 0003)
```

Примечания:

Если указанная страница доступна отладчику DEBUG, то DEBUG выводит сообщение об успешном создании канала, в противном случае выводится сообщение об ошибке.

DEBUG: XD (Deallocation EXpanded Memory)

Отключение канала EMS.

Для использования расширенной памяти вы должны инсталлировать драйвер расширенной памяти перед использованием отладчика DEBUG.

Этот драйвер должен удовлетворять стандарту LIM EMS 4.0.

Формат команды:

XD *канал*

Параметр:

канал - указание номера отключаемого канала расширенной памяти.

Пример использования:

Для отключения канала номер 0003 введите команду:

```
XD0003
```

Если операция отключения прошла успешно, DEBUG выводит следующее сообщение:

```
Handle 0003 deallocated
(Канал 0003 отключен)
```

DEBUG: XM (Map Expanded Memory)

Изменение логических страниц EMS, принадлежащих указанным каналам, на физические страницы дополнительной памяти (XMS). Для использования расширенной памяти вы должны инсталлировать драйвер расширенной памяти перед использованием отладчика DEBUG. Этот драйвер должен удовлетворять стандарту LIM EMS 4.0.

Формат команды:

XM *страница1 страница2 канал*

Параметры:

страница1 - указание номера логической страницы EMS, которую вы хотите изменить на физическую страницу в параметре *страница2*.

страница2 - указание номера физической страницы дополнительной памяти (XMS), на которую будет изменена логическая страница расширенной памяти.

канал - указание номера канала для работы с EMS.

Пример использования:

```
XM 5 2 0003
```

При успешном выполнении этой команды DEBUG выводит на дисплей следующее сообщение:

```
Logical page 05 mapped to phisical page 02
(Логическая страница 05 изменена на физическую страницу 02)
```

DEBUG: XS (Display Expanded Memory Status)

Вывод на дисплей информации о EMS.

Для использования расширенной памяти вы должны инсталлировать драйвер расширенной памяти перед использованием отладчика DEBUG.

Этот драйвер должен удовлетворять стандарту LIM EMS 4.0.

Формат команды:

XS

Пример использования:

Для получения на экране дисплея информации о расширенной памяти EMS введите:

```
XS
```

При успешном выполнении этой команды DEBUG выведет сообщение в следующем формате:

```
Handle 0000 has 0000 pages allocated
Handle 0001 has 0002 pages allocated
Physical page 00 = Frame segment C000
Physical page 01 = Frame segment C400
Physical page 02 = Frame segment C800
Physical page 03 = Frame segment CC00
2 of a total 80 EMS pages have been allocated
2 of a total FF EMS handles have been allocated
```

Полученный отчет содержит информацию о доступных страницах расширенной памяти, расположении физических страниц EMS и их адресах, общем количестве страниц EMS и каналов для работы с расширенной памятью.

(!) DEFRAG

Запуск утилиты дефрагментации файлов. Эта утилита используется для оптимизации файловой системы. Программа допускает управление ею как с помощью удобного пользовательского интерфейса, так и с помощью обычного для DOS задания параметров командной строки.

По своим возможностям и способам управления программа повторяет интерфейс известной программы подобного класса Speed Disk из пакета Norton Utilites, что вполне понятно, так эта программа специально разработана для MS-DOS 6.0 компанией Symantec Corporation (изготовитель программы Speed Disk).

Команда внешняя. Файл DEFRAG.EXE. Внешний вид основного меню программы представлен на рисунке 8.2.

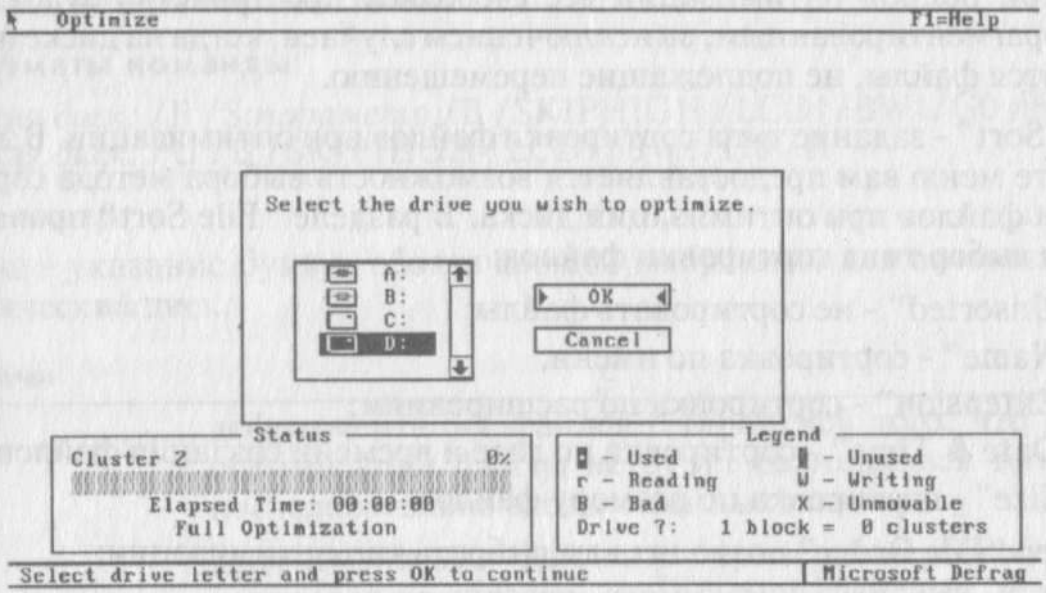


Рис. 8.2

Управление осуществляется с помощью манипулятора типа "мышь", либо с помощью меню программы. Для входа в меню используется клавиша Alt.

После запуска DEFRAG программа попросит вас выбрать логический диск для оптимизации из числа доступных. Далее утилита проанализирует файловую структуру выбранного диска и предложит метод оптимизации. Вы можете использовать предложенный метод, но также можете изменить параметры оптимизации, войдя в меню "Optimize". После этого программа займется оптимизацией файловой системы.

Вы всегда можете прервать ее выполнение с помощью клавиши Esc или нажатием на одну из клавиш "мыши". Меню "Optimize" содержит пункты:

"Begin Optimization Alt-B" - начало оптимизации.

"Drive" - выбор логического диска. Этот пункт меню аналогичен подменю, появляющемуся при старте программы.

"Optimization Method " - выбор метода оптимизации. После выбора этого пункта вы можете выбрать в открывшемся подменю 2 метода оптимизации:

1. "Full optimization" - полная оптимизация файловой системы с размещением файлов и каталогов в последовательных кластерах.
2. "Unfragment files only" - оптимизация только фрагментированных файлов без записи их в последовательные сектора. Программа в

этом варианте работы по возможности постарается освободить как можно больше нефрагментированного дискового пространства. При полной оптимизации все свободное пространство будет нефрагментированным, за исключением случаев, когда на диске имеются файлы, не подлежащие перемещению.

"File Sort" - задание типа сортировки файлов при оптимизации. В этом пункте меню вам предоставляется возможность выбора метода сортировки файлов при оптимизации диска. В разделе "File Sort" производится выбор типа сортировки файлов:

- "Unsorted" - не сортировать файлы;
- "Name" - сортировка по имени;
- "Extension" - сортировка по расширениям;
- "Date & Time" - сортировка по дате и времени создания файлов;
- "Size" - сортировка по размеру файлов.

Раздел "File Order" позволит вам выбрать тип оптимизации:

- "Ascending" - преобладающая оптимизация;
- "Descending" - непреобладающая оптимизация;

"Map legend" - объяснение условных обозначений, используемых при работе утилиты DEFRAG ". Условные обозначения:

"Disk Space used by files" - дисковая поверхность, используемая файлами;

"Disk Space optimized only" - дисковая поверхность, которая оптимизирована;

"Unized disk space" - дисковая поверхность, не используемая файлами;

X "Disk space used by first that will not be moved" - дисковая поверхность, которую занимают файлы, не подлежащие перемещению;

B "Bad disk space " - дисковая поверхность, которую занимают дефектные сектора, обнаруженные при форматировании диска;

r "Disk space that is being read" - дисковая поверхность, с которой производится считывание информации;

W "Disk space that is being written" - дисковая поверхность, на которую в данный момент производится запись информации;

"About Defrag" - информация о программе DEFRAG. В этом пункте меню отражена информация об авторских правах на эту программу фирмы Symantec Corporation и технологии Norton Utility, использованной при разработке этой программы;

"Exit Alt-X" - выход из программы DEFRAG в среду MS-DOS.

Для работы с DEFRAG при помощи командной строки используется способ, который описан ниже.

Форматы команды:

defrag *диск:* /F /S:*параметр* /B /SKIPHIGH /LCD|/BW|/G0 /H
defrag *диск:* /U /B /SKIPHIGH /LCD|/BW|/G0 /H

Параметр:

диск: - указание буквы, обозначающей выбранный для оптимизации логический диск.

Ключи

/F	дефрагментация файлов с гарантией того, что после оптимизации файлы не будут размещаться только в последовательно расположенных кластерах;
/U	дефрагментация файлов с перемещением свободных кластеров из области, занимаемой файлами, в другое место на диске;
/S	задание метода сортировки файлов. Этот ключ включает в себя *параметр*. При использовании этого ключа файлы будут отсортированы в своих каталогах в соответствии с методом, который задан ключом *параметр*. Если параметр *диск:* не указан, то сортировка будет осуществляться для текущего диска. Ключ *параметр* может принимать значения, которые указаны ниже: "N" - Задание сортировки файлов по имени; "N-" - Сортировка файлов по имени задается в обратном порядке (от Z до A); "E" - Сортировка файлов по расширению файлов; "E-" - Задание сортировки файлов по расширению файлов в обратном порядке; (от Z до A); "D" - Сортировка по дате и времени создания файлов; "D-" - Сортировка файлов по дате и времени их создания в обратном порядке; "S" - Сортировка файлов по их размеру, начиная от файлов с меньшей длиной до файлов с большей длиной; "S-" - Сортировка файлов по их размеру, начиная от файлов с большей длиной к файлам с меньшей длиной;

/B	Организация перезагрузки вашего компьютера после того, как все файлы будут реорганизованы на диске;
/SKIPHIGH	загрузка утилиты DEFRAG в область основной памяти. По умолчанию DEFRAG загружается в область старшей памяти, если она является доступной для прикладных программ;
/LCD	запуск программы DEFRAG на компьютере, оснащенном жидкокристаллическим дисплеем;
/BW	запуск программы DEFRAG с отображением информации на дисплее, находящимся в черно-белом режиме;
/GO	запрещение использования режима графического диалога DEFRAG с использованием указателя "мыши" в виде стрелки и графического режима работы дисплея (только для мониторов EGA/VGA и выше);
/H	разрешение перемещения утилитой DEFRAG "скрытых" файлов.

Примеры использования:

Для того, например, чтобы загрузить утилиту DEFRAG в область основной памяти, организовать сортировку файлов по времени их создания, причем необходимо, чтобы файлы, которые созданы позднее, находились ближе к началу диска, введем для диска C: следующую команду:

```
defrag c: /F /SD- /SKIPHIGH
```

После выполнения этой команды диск C: будет полностью оптимизирован утилитой DEFRAG, которая также осуществит сортировку файлов согласно указанному методу.

Примечания:

- Используя утилиту DEFRAG, вы не можете оптимизировать сетевые диски, созданные утилитой INTERKLNK.
- Различия отчетов программ DEFRAG и CHKDSK Информация о файловой структуре, выдаваемая утилитой DEFRAG, отличается от информации о файловой структуре команды CHKDSK. Эти различия проявляются в следующем. DEFRAG не делает различий между пользовательскими и "скрытыми" файлами. CHKDSK же различает файлы с различными атрибутами. В процессе своей работы DEFRAG производит перемещение каталогов, CHKDSK этих операций не выполняет. Кроме этого, DEFRAG не анализирует метку диска, а CHKDSK делает это.

- Программу DEFRAG нельзя запускать, работая с Microsoft Windows, так как это может привести к необратимым последствиям и краху вашей системы.
- DEFRAG после окончания своей работы возвращает в MS-DOS свой код завершения, который может использоваться в пакетных файлах (команда IF ERRORLEVEL) или выполняемых модулях. Коды завершения утилиты DEFRAG приведены ниже:

0	дефрагментация файлов прошла успешно;
1	в процессе выполнения программы произошла ее внутренняя ошибка;
2	диск не содержит свободных кластеров. Для работы DEFRAG минимум необходимо наличие 1 свободного кластера;
3	пользователь использовал нажатие комбинации клавиш Ctrl+C для остановки работы программы;
4	в процессе работы произошла глобальная ошибка;
5	DEFRAG обнаружил ошибку в процессе чтения кластера;
6	DEFRAG обнаружил ошибку в процессе записи данных в кластер;
7	обнаружена ошибка, связанная с нарушением целостности файловой структуры диска. Для коррекции ошибок необходимо использовать команду CHKDSK с ключом /F;
8	обнаружена ошибка, связанная с расположением программ и данных в оперативной памяти;
9	недостаточно памяти для работы дефрагментатора.

- Если на вашем компьютере имеется пакет Norton Utilites версии 6.0 и выше, вы можете также изменять вид меню на экране дисплея с использованием программы NUCONFIG (в оболочке пакета ее имя - Configuration). Для выполнения вышеуказанной операции достаточно выбрать в программе Configuration пункт "Video & Mouse", где и установить необходимые режимы работы дисплея.

DEL (ERASE)

Удаление одного или нескольких указанных файлов. Команда внутренняя.

Формат команды:

del *диск:\маршрут\имя_файла* /P
erase *диск:\маршрут\имя_файла* /P

Параметры:

диск:\маршрут\имя-файла - указание полного пути доступа к удаляемому файлу или группе файлов. Если этот параметр не указан, то считается, что удаляемый файл (файлы) находятся в текущем каталоге.

Ключ

/P при использовании этого ключа DOS будет спрашивать перед удалением каждого из указанных файлов разрешения пользователя.

Примеры использования:

Допустим, вам необходимо удалить файл с именем CAT.TMP, находящийся в каталоге TEST логического диска C:. Для выполнения этой операции введите одну из следующих команд:

```
del c:\test\cat.tmp
erase c:\test\cat.tmp
```

Для удаления всех файлов в каталоге с именем TEST на диске C: используйте одну из приведенных ниже команд:

```
del c:\test
del c:\test\*.*
```

Примечания:

- При использовании ключа /P DEL будет выводить на дисплей имя файла, подлежащего удалению, и выводить сообщение в следующем формате:

  (file_name), Delete (Y/N)?
  (имя-файла, Удалить (Д/Н)?)

  Нажмите клавишу Y для удаления указанного файла или клавишу N для отмены его удаления и вывода на дисплей имени следующего удаляемого файла. Вы можете прервать удаление одного или группы файлов, воспользовавшись комбинацией клавиш Ctrl-C.

- При удалении всех файлов в указанном каталоге вы можете указывать следующие групповые шаблоны: ? и *. Однако, когда вы будете использовать групповые шаблоны в команде DEL, DOS попросит вас еще раз убедиться в правильности вашего решения. Допустим, вы ввели следующую команду:

  ```
  del *.*
  ```

  После ввода этой команды на экране появится следующее сообщение:

  All files in directory will be deleted! Are you sure (Y/N)?
  (Все файлы в этом каталоге будут уничтожены! Вы уверены (Д/Н)?)

Нажав клавишу Y и затем Enter, вы удалите все файлы в текущем каталоге. Если будет нажата клавиша N и затем Enter, то команда DEL выполнена не будет.

- При использовании команды DEL с групповыми шаблонами удобно вначале просмотреть содержимое выбранного каталога с помощью команды DIR. Подробности ее использования рассматриваются далее.
- Удаленные командой DEL файлы подлежат восстановлению утилитой DOS UNDELETE, либо любой другой программой аналогичного класса. Однако восстановление удаленных файлов будет возможно только в том случае, если после использования команды DEL не производилась запись информации на диск, содержащий удаленные файлы. Подробности использования утилиты UNDELETE можно посмотреть в ее описании.

(!) DELTREE

Удаление каталогов, включая все файлы и подкаталоги, входящие в удаляемый каталог. Команда внешняя. Файл DELTREE.EXE

Формат команды:

deltree /Y *диск:\маршрут*

Параметры:

диск:\маршрут - указание имени каталога выбранного логического диска, который необходимо удалить. Команда DELTREE будет в процессе своей работы удалять все файлы, находящиеся как в указанном каталоге, так и в подчиненных ему.

Ключ

/Y	при указании этого ключа утилита DELTREE при удалении указанного каталога не будет спрашивать разрешения пользователя, что она делает по умолчанию.

Пример использования:

Для удаления каталога TEMP на диске C:, включая удаление всех входящих в него каталогов и файлов, наберите следующую команду:

```
deltree c:\temp
```

Примечания:

- Команда DELTREE удаляет все файлы в указанном каталоге, подчиненные подкаталоги и все файлы, входящие в них не обращая внимания на атрибуты файлов и каталогов.

- Если DELTREE произвела успешное удаление каталога, то она возвращает свой код завершения, равный 0, для дальнейшего использования его, например, в параметре ERRORLEVEL командных файлов.
- В команде DELTREE допускается использования групповых шаблонов для указания имен удаляемых каталогов и файлов. Однако следует напомнить, что перед удалением каталогов и файлов не лишним будет просмотреть их содержимое командой DIR или другими средствами.

DIR

Вывод на дисплей списка файлов и подкаталогов, находящихся в указанном пользователем каталоге.

Когда команда DIR задается без параметров и ключей. По умолчанию на дисплей выводятся:

- метка диска и его серийный номер;

- список файлов и подкаталогов текущего каталога.

Данные о каждом файле (подкаталоге) выводятся в одну строку, содержащую сведения об имени файла (подкаталога), его расширении, размере файлов в байтах, дате и времени последней модификации файлов, а также об общем объеме дискового пространства, пространства, занятого файлами в текущем каталоге и на всем диске и общем объеме свободного дискового пространства. Команда внутренняя.

Формат команды:

dir *диск:\маршрут\имя_файла* /P /W /A:*атрибуты* /O:*тип_сортировки* /S /B /L /C

Параметры:

диск:\маршрут - указание имени диска и каталога, листинг которых вы желаете получить;

имя_файла - указание имени одного или группы файлов, характеристики которого(рых) необходимо посмотреть.

Ключи

/P	вывод содержимого каталога по страницам. После заполнения одной страницы листингом каталога DOS будет ожидать нажатия любой клавиши для показа очередной страницы листинга. Страница, отображаемая DIR, в этом случае имеет вид стандартного экрана дисплея.

/W - вывод листинга каталога в ширину. Для вывода информации о файлах используется 5 колонок, содержащих только самую необходимую информацию (имя файла).

/A:*атрибуты* задание режима вывода на дисплей только тех файлов, атрибуты которых совпадают с указанными в команде DIR. Если этот ключ опущен, команда DIR выводит на дисплей имена всех файлов, за исключением файлов, имеющих атрибуты "системный" и "скрытый". Параметр *атрибуты* отделяется от ключа /A символом ":", но этот прием не является обязательным. Для включения в список файлов указанного каталога, файлов, которые не показываются DIR в обычном режиме, используйте с ключом /A параметры *атрибуты*, обозначения которых и функции, ими реализуемые, приведены ниже:

"H" - показывать "скрытые" файлы;
"-H" - показывать файлы, не являющиеся "скрытыми";
"S" - показывать "системные" файлы;
"-S" - показывать файлы, не являющиеся "системными";
"D" - включать в отчет, только подчиненные каталоги;
"-D" - показывать в листинге только имена файлов;
"A" - включать в листинг каталогов файлы, имеющие атрибут "архивный";
"-A" - включать в отчет файлы, не имеющие изменений со времени последнего резервного копирования;
"R" - включать в отчет файлы с атрибутом "только для чтения";
"-R" - включать в отчет файлы, не имеющие атрибута "только для чтения".

/O:*тип_сортировки* задание типа сортировки имен файлов и каталогов при выводе их на экран дисплея. Если этот ключ опущен, DIR выводит на дисплей имена файлов в том порядке, в котором они записаны в каталогах. Если вы используете этот ключ без параметра *тип_сортировки*, то DIR по умолчанию отсортирует имена файлов и каталогов в листинге каталога в алфавитном порядке. Для использования параметра *тип_сортировки* с ключом /O он должен отделяться от ключа символом ":", хотя допускается этот символ не использовать, а записывать параметр слитно с ключом. Параметр *тип_сортировки* может принимать следующие значения:

"N" - сортировка по именам файлов в алфавитном порядке;
"-N" - сортировка по именам файлов в обратном алфавитном порядке (от Z до A);
"E" - сортировка по расширениям файлов в алфавитном порядке;

"-E" - сортировка по расширениям файлов в обратном алфавитном порядке (от Z до A);
"D" - сортировка по дате и времени создания файла, начиная от файлов с более ранними датой и временем создания к файлам с более поздними датой и временем создания;
"-D" - сортировка по дате и времени создания файлов в обратном порядке, начиная от файлов с более поздней датой и временем создания к файлам с более ранними датой и временем создания;
"S" - сортировка файлов по их размеру, начиная от файла с наименьшим размером до файла с наибольшим размером;
"-S" - сортировка файлов по их размеру в обратном порядке, начиная от файла с наибольшей длиной к файлу с наименьшей длиной;
"G" - сортировка с показом списка подчиненных каталогов перед списком файлов;
"-G" - сортировка с показом списка подчиненных каталогов после списка файлов;
"C" - сортировка файлов по коэффициенту сжатия, начиная от файла с наименьшим коэффициентом сжатия к файлу с наибольшим коэффициентом сжатия. Этот параметр применяется при работе команды DIR с "сжатыми" дисками;
"-C" - сортировка файлов по коэффициенту сжатия в обратном порядке, начиная от файла с наибольшим коэффициентом сжатия к файлу с наименьшим коэффициентом сжатия. Этот параметр применяется при работе команды DIR с "сжатыми" дисками.

/S — этот ключ заставляет команду DIR включать в свой листинг файлы, находящиеся в подчиненных каталогах, а не только файлы, находящиеся в указанном каталоге;

/B — при использовании этого ключа данные о каталоге или файле отображаются в одну строку, включая расширение файла. Этот ключ предписывает DIR не выводить информацию об общих характеристиках диска. Он не может быть совместно использован с ключом /W;

/L — употребление этого ключа в команде DIR приводит к тому, что в листинге каталогов имена каталогов и файлов не будут отсортированы, и вся информация будет отражена только с использованием прописных букв;

/C|H — вывод на дисплей информации, отражающей коэффициент сжатия файлов, обработанных ранее утилитой Double Space. Сжатые файлы при работе утилиты Double Space используют для размещения файлов кластеры размером 8 K. Если вместо ключа /C используется ключ /H, то коэффициент сжатия файлов на "сжатых" дисках рассчитывается, исходя из размеров кластера на "несжатом" ("горячем") диске. Ключ /C[H] игнорируется, если используются ключи /W или /B.

Примеры использования:

Для вывода на дисплей списка всех файлов и подчиненных каталогов, включая "скрытые" и "системные" файлы, введите команду:

```
dir /A
```

Если вам необходимо вывести на дисплей содержимое всех каталогов текущего логического диска, причем необходимо, чтобы информация выводилась в ширину, постранично и без сортировки списка файлов, введите следующую команду:

```
dir /S /W /O /P
```

DIR выведет на дисплей обозначение корневого каталога, имена подчиненных ему каталогов и имена файлов, находящихся в корневом каталоге вместе с их расширениями. Далее DIR выведет последовательно листинги каталогов, подчиненных корневому.

В том случае, если вам необходимо, чтобы команда DIR выполнила операцию, аналогичную ранее приведенной, но так, чтобы список подчиненных каталогов выводился после списка файлов корневого каталога, введите команду:

```
dir /S /W /O /P /A:-D
```

Для печати листинга текущего каталога на принтере возможно использование команды DIR в той форме, в какой она приведена ниже:

```
dir > prn
```

Так в этой команде DIR использовано имя устройства PRN, то листинг текущего каталога будет распечатан на принтере, подключенном к порту LPT1 (PRN синоним LPT1). Если же вам необходимо распечатать информацию, выдаваемую командой DIR на принтере, подключенном к другому параллельному порту, например LPT2, то необходимо использовать вместо PRN имя устройства LPT2.

Команда DIR предоставляет пользователям возможность направлять отчеты, выдаваемые DIR не на принтер, а в указанный файл. Для примера, направим выходной листинг команды DIR в файл с именем DIR.DOC, находящийся в каталоге RECORDS текущего диска:

```
dir > \records\dir.doc
```

В том случае, когда файл DIR.DOC не существует, MS-DOS создаст его. Однако если не существует указанный каталог, MS-DOS выведет на дисплей следующее сообщение:

```
File creation error
(Ошибка создания файла)
```

Для вывода на дисплей листинга всех файлов с расширением .TXT на диске C:\ во всех каталогах используйте команду:

```
dir c:\*.txt /W /O /S /P
```

В результате выполнения этой команды на дисплей будет выведен список файлов в ширину, в алфавитном порядке, в каждом каталоге диска C: и постранично.

Примечания:

- В ключе /A команды DIR допускается использование нескольких параметров. Допустим, вам необходимо, чтобы команда DIR вывела на дисплей список файлов, имеющих атрибут "только для чтения", но не являющихся "скрытыми". Для решения этой задачи вы должны использовать в ключе /A параметры R и -H совместно, например, записав ключ /A в виде /A:R-H или /AR-H.
- Ключ /O команды DIR аналогично ключу /A этой же команды и допускает совместное использование нескольких своих параметров. Следует только помнить о порядке сортировки файлов при выводе их списка на дисплей. DIR при сортировке имен файлов будет руководствоваться критериями сортировки в том порядке, в каком они записаны в ключе /O. В качестве примера рассмотрим вариант, когда ключ /O использует параметры E и -S, записанные в виде /O:E-S или /OE-S. В данном случае команда DIR отсортирует файлы в алфавитном порядке по их расширениям, начиная от файла с наибольшей длиной к файлу с наименьшей длиной.
- При работе команда DIR использует форматы даты и времени, заданные пользователем в файле конфигурации системы CONFIG.SYS. Если команда COUNTRY файла CONFIG.SYS не используется, то используются форматы даты и времени, принятые в США.
- При работе с символами ">" и "|" для перенаправления выходной информации команды DIR на другое устройство, взамен стандартного, или на вход другой команды DOS используйте ключи /A:-D и /B для улучшения вида выходной информации DIR. Вы также можете использовать ключ /B совместно с ключом /S для вывода командой DIR информации о всех каталогах, подчиненных указанному и всех файлах, располагающихся в просматриваемых каталогах. В этом случае DIR выведет в листинг имя диска, имена каталогов, имена файлов с их расширениями, маршруты доступа к каждому из каталогов.
- Для повышения скорости работы команды DIR в операциях перенаправления и конвейеризации установите для переменной TEMP окружение в вашем файле AUTOEXEC.BAT. В качестве параметра

для переменной TEMP укажите полный идентификатор файла, в который планируется запись выходной информации команды DIR.

- ОС MS-DOS предоставляет пользователю возможность предварительно установить параметры и ключи команды DIR, включив в файл AUTOEXEC.BAT команду SET для установки окружения переменной с именем DIRCMD. Описанная операция позволит выполнять команду с параметрами и ключами, воспринимаемыми DOS как заданными по умолчанию.

  Для примера, присвоим переменной DIRCMD значение /W, что заставит команду DIR выводить свой листинг в ширину, если не указаны другие параметры, изменяющие вид выходного листинга DIR. Для присвоения переменной DIRCMD указанного значения введем в файл AUTOEXEC.BAT следующую команду:

  ```
  set dircmd = /W
  ```

  Если же вам в процессе работы необходимо однократно изменить установки параметров и ключей DIR, заданные по умолчанию, то эта операция выполняется обычным способом, вводом команды DIR с измененными ключами или параметрами, как в следующем примере:

  ```
  dir /-W
  ```

  Пользователь может также изменить установки по умолчанию DIR, изменив окружение переменной DIRCMD с помощью команды DOS SET. Измененные установки команды DIR будут действовать как установки по умолчанию для всех команд DOS, до момента, пока окружение переменной DIRCMD не будут снова изменено или компьютер не будет перезагружен. Для очистки переменной DIRCMD от всех установок используйте команду:

  ```
  set dircmd =
  ```

  Если же вам необходимо посмотреть текущие установки окружения переменной DIRCMD, введите просто команду SET:

  ```
  set
  ```

  MS-DOS выведет на дисплей список всех переменных системного окружения, в числе которых будет значиться переменная DIRCMD с ее ключами и параметрами. Подробности использования команды SET даны в ее описании.

- Команда DIR поддерживает "сжатые" диски под управлением DBLSPACE. Подробности работы со сжатыми дисками приведены в описании утилиты DBLSPACE.

DISKCOMP

Сравнение содержимого двух гибких дисков. Эта команда производит сравнение содержимого гибких дисков по сторонам и дорожкам. DISKCOMP определяет номер стороны и дорожки гибкого диска в которых имеются расхождения в данных. В качестве диска, который является образцовым, в этой команде выступает диск, указанный первым. Команда внешняя. Файл DISKCOMP.COM.

Формат команды:

diskcomp *диск1: диск2:* /1 /8

Параметры:

диск1: - указание буквы, обозначающей дисковод, в котором находится первый из сравниваемых дисков.

диск2: - указание буквы, обозначающей дисковод со вторым сравниваемым диском.

Ключи

/1	сравнению будет подвергаться только одна сторона гибких дисков. Если диски двусторонние, и дисковод может читать двусторонние диски, то будет сравниваться четная сторона дисков;
/8	сравниваться будут только 8 секторов на дорожке. Если диски имеют 8 или 15 секторов, то будут сравниваться только четные сектора.

Примеры использования:

Допустим ваш компьютер имеет лишь один дисковод A, а вам необходимо сравнить 2 гибких диска. Введите в этом случае команду:

```
diskcomp a: a:
```

DISKCOMP попросит вас попеременно вставлять каждый диск и будет информировать о ходе сравнения дискет.

Примечания:

- Команда DISKCOMP работает только с гибкими дисками. Вы не можете использовать DISKCOMP для работы с винчестером. Если вы укажете в качестве параметров *диск1:* и *диск2:* команды DISKCOMP логические диски винчестера, то DOS выведет на дисплей следующее сообщение:

```
Invalid drive specification
Specified drive does not exist
```

or is non-removable
(Неправильное имя диска
Указанный диск не существует
или является несъемным)

В случае успешного сравнения 2 гибких дисков на дисплее появится следующее сообщение:

Compare OK
(Сравнение OK)

Если же в процессе сравнения обнаружена ошибка, DISKCOMP выводит на дисплей сообщение следующего формата:

Compare error on
side 1, track 2
(Ошибка сравнения на
стороне 1, дорожке 2)

Когда утилита DISKCOMP завершает сравнение, на дисплей выводится следующее сообщение:

Compare another diskette (Y/N)?
(Сравнить другие дискеты (Д/Н)?)

Если вы нажмете клавишу Y, DISKCOMP попросит вас вставить дискеты для следующего сравнения, если пользователь нажимает клавишу N, работа утилиты DISKCOMP прекращается. В процессе работы утилита DISKCOMP игнорирует серийный номер диска при сравнении.

- При отсутствии в команде DISKCOMP параметра *диск:2* утилита использует в качестве второго дисковода текущий диск. Если текущим диском является диск, указанный в параметре *диск1*, то сравнение происходит аналогично ранее описанному примеру использования команды DISKCOMP.
- Утилита DISKCOMP не осуществляет сравнение разных типов гибких дисков, например одно- и двухсторонних дисков или дисков с двойной и высокой плотностью записи. Если же вы попытаетесь сравнить два диска разных типов, на дисплее появится сообщение:

Drive types or diskette types not compatible
(Типы дисководов или дискет не совпадают)

- Утилита DISKCOMP не позволяет организовывать работу с сетевыми дисками, а также дисками, созданными или поддерживаемыми командой SUBST. В том случае, если вы попытаетесь использовать DISKCOMP для работы с сетевыми дисками или созданными командой SUBST, DISKCOMP введет следующее сообщение об ошибке:

  Comparing an original disk with a copy
  (Сравнение оригинального диска вместе с копированием)

  Когда вы будете использовать DISKCOMP дальше, после того как вы скопируете файлы командой COPY, DISKCOPY выведет сообщение:

  Compare error on
  side 0, track 0
  (Ошибка сравнения на
  стороне 0, дорожке 0)

  Эта ошибка возникает в том случае, если на диске уже находились файлы. Хотя команда COPY позволяет дублировать информацию, это свойство команды COPY не гарантирует вам, что при новой записи информации она будет занимать то же местоположение на диске, что и ранее записанная информация. Дополнительную информацию по сравнению двух файлов вы сможете получить в описании команды FC.

- Команда DISKCOMP по окончании своей работы возвращает DOS свои коды завершения для дальнейшего их использования, например, в команде IF ERRORLEVEL пакетных файлов. Коды завершения DISKCOMP и их значение приведены далее:

  0 сравниваемые диски идентичны;
  1 в структуре данных сравниваемых дисков обнаружены различия;
  2 пользователь прервал выполнение программы, нажав комбинацию клавиш Ctrl+C;
  3 во время выполнения программы обнаружена критическая ошибка;
  4 обнаружена ошибка инициализации.

DISKCOPY

Копирование содержимого одного гибкого диска на другой гибкий диск. DISKCOPY записывает информацию поверх старой на назначенный диск (диск-приемник), независимо от того имелась там ранее информация или нет.

Эта утилита перед началом работы сама определяет тип дисковода и гибкого диска, в него установленного.

Команда внешняя. Файл DISKCOPY.COM.

Формат команды:

diskcopy *диск1: диск2:* /1 /V

Параметры:

диск1: - указание буквы, обозначающей диск с которого, будет осуществляться копирование информации (диск-источник).

диск2: - указание буквы, обозначающей диск-приемник.

Ключи

/1	этот ключ предписывает DISKCOPY копировать только первую сторону диска;
/V	задание режима верификации (проверки) правильности записи информации после копирования. Используйте этот ключ при копировании очень ценной и важной информации.

Примечания:

- Утилита DISKCOPY не работает только со съемными дисками, каковыми являются дискеты. Вы не сможете применить эту команду для работы с жестким диском. Если укажите в качестве параметров *диск:1* и *диск2:* логические диски винчестера, то DISKCOPY выведет на дисплей следующее сообщение об ошибке:

  Invalid drive specification
  Specified drive does not exist
  or is non-removable
  (Неправильное имя диска
  Указанный диск не существует
  или является несъемным)

- Во время работы DISKCOPY периодически просит пользователя вставлять диск-источник и диск-приемник, если емкость диска не позволяет загрузить его весь в оперативную память. Пользователю необходимо только менять дискеты и нажимать далее любую клавишу для продолжения процесса копирования. После того, как копирование будет завершено, DISKCOPY выведет сообщение:

  Copy another diskette (Y/N)?
  (Скопировать другую дискету (Д/Н)?)

  Если будет нажата клавиша Y, DISKCOPY попросит вас вставить новый диск-источник, и копирование будет продолжено; если вы нажмете N - работа DISKCOPY завершится.

- В том случае, если копирование информации производится на неформатированный диск DISKCOPY автоматически проведет фор-

матирование диска-приемника. При этом число сторон, дорожек и секторов на дорожках будет соответствовать значениям, установленным для диска-источника. В этом случае на дисплее появится сообщение:

Formatting while copying
(Форматируется, пока копируется)

- Утилита DISKCOPY не позволяет копировать информацию в случае, если емкость диска-источника выше емкости диска-приемника. Когда ваш компьютер обнаружил это различие, DISKCOPY выводит на дисплей сообщение следующего формата:

Drive types or diskette types not compatible
(Типы дисководов или типы дискет несовместимы)

- Если диск-источник имеет серийный номер, DISKCOPY присваивает диску-приемнику новый серийный номер и выводит его на дисплей, когда операция копирования завершена.
- Когда в команде DISKCOPY вы не используете параметр *диск2:*, DOS использует в качестве диска-приемника текущий диск. Если оба параметра команды DISKCOPY опущены, то DISKCOPY использует как для диска-источника, так и для диска-приемника текущий диск. В случае, когда диском-источником и диском-приемником выступает один и тот же дисковод, DISKCOPY будет попеременно просить вставлять в дисковод диск-источник и диск-приемник.
- Использование команды DISKCOPY приводит к тому, что на диске-приемнике размещение файлов будет абсолютно аналогично размещению файлов на диске-источнике. Таким образом, если диск-источник содержит фрагментированные файлы, то эти фрагментированные файлы будут перенесены и на диск-приемник. Для решения проблемы фрагментированных файлов используйте для копирования файлов на диск-приемник, команды COPY или XCOPY, которые будут записывать файлы на диске-приемнике в непрерывные кластеры.
- Команду DISKCOPY удобно использовать для создания системных дисков. Если вы попытаетесь создать системный диск, скопировав на него файлы ОС при помощи команд COPY или XCOPY может возникнуть ситуация, когда системные файлы MS-DOS будут располагаться не в отведенных для них местах.

- Команда DISKCOPY после окончания своей работы возвращает в ОС свои коды завершения для дальнейшего их использования в пакетных файлах или выполняемых модулях. Коды завершения команды DISKCOPY и их значения приведены ниже:

  0 операция копирования прошла успешно;
  1 обнаружена нефатальная ошибка записи-чтения;
  2 прерывание программы пользователем по нажатию клавиш Ctrl-C;
  3 обнаружена критическая ошибка;
  4 в процессе работы обнаружена ошибка инициализации.

- Рекомендуется после выполнения команды DISKCOPY провести сравнение диска-источника и диска-приемника при помощи команды DISKCOMP. Подробности использования команды DISKCOMP обсуждаются в описании этой команды.

DOSKEY

Расширение возможностей управления командной строкой DOS, создание макроопределений для команд, редактирование командной строки DOS.

DOSKEY является резидентной в памяти программой. После загрузки она занимает в оперативной памяти примерно 3 K.

Формат команды:

doskey /REINSTALL /BUFSIZE = *размер* /MACROS /HISTORY /INSERT|/OVERSTRIKE *макроимя* = *текст*

Для запуска программы DOSKEY с установками, задаваемыми по умолчанию, просто введите:

DOSKEY

Параметр:

макроимя = *текст* - использование этого параметра позволяет создать макроопределение для одной или нескольких команд MS-DOS (макрос DOSKEY). Под *макроопределением* понимается некоторая совокупность символов, задающих, например, одну или несколько команд DOS с их параметрами, но обозначаемая одним словом.

Если вам необходимо выполнить последовательность команд DOS, которая задана макроопределением, то вам достаточно лишь воспользоваться макроопределением, введя его с клавиатуры, вместо того чтобы последовательно выполнять команды DOS.

Команда внешняя. Файл DOSKEY.COM.

Ключи

/REINSTALL	загрузка в память новой копии программы DOSKEY, четной если одна копия уже находится в памяти. В последнем случае ключ /REINSTALL также производит очистку буфера DOSKEY;
/BUFSIZE= *размер*	указание в байтах размера буфера, используемого DOSKEY для хранения команд и макросов DOSKEY. По умолчанию размер буфера равен 512 байт. Минимально допустимое значение для параметра *размер* в ключе /BUFSIZE равно 256 байт;
/MACROS	вывод на дисплей всех макросов DOSKEY. С этим ключом допустимо использование символа перенаправления вывода ">" совместно с ключом /MACROS, для направления выходной информации DOSKEY в файл. Допускается использование вместо аббревиатуры /MACROS ключа /M.
/HISTORY	вывод на дисплей списка всех запомненных DOSKEY команд. С этим ключом возможно использование символа ">" для перенаправления вывода информации в файл. Вместо ключа /HISTORY можно использовать ключ /H.
/INSERT|/ OVERSTRIKE	указание способа ввода нового текста в старый текст. Если использован ключ /INSERT, то при вводе нового текста в старый, старый текст будет раздвигаться, начиная от позиции, с которой начался ввод нового текста (так, как если бы вы нажали клавишу Insert). Если же вы используете ключ /OVERSTRIKE, новый текст будет записываться поверх старого;

Примеры использования:

Ключи /MACROS и /HISTORY обычно используются для создания командных файлов, содержащих записанные макроопределения и команды, как это сделано в примере ниже. Файл MACINIT.BAT будет включать в себя все макросы DOSKEY после ввода команды:

```
doskey /MACROS > macinit.bat
```

Для дальнейшего использования файла MACINIT.BAT отредактируйте входящие в него команды DOSKEY для каждой строки макроопределений. Для создания командного файла TMP.BAT, содержащего все команды DOS, которые запомнила утилита DOSKEY, введите следующую команду:

```
doskey /HISTORY > tmp.bat
```

Если вам необходимо создать макрос для нескольких команд используйте символы "$T" для разделения команд между собой, например:

```
doskey tx=cd\temp$tdir /W $*
```

В приведенном примере макроопределению TX ставится в соответствие следующая последовательность команд DOS:

- выполняется команда смены текущего каталога на C:\TEMP;
- выводится в ширину листинг каталога TEMP командой DIR/W.

Символы "$*" используются для окончания строки макроопределения и добавления ключей команды DIR. При задании макроопределений в них допустимо использование формальных параметров, как это сделано в следующем примере:

```
doskey mc=md $1$tcd $1
```

В этом примере в строке макроопределения используется формальный параметр, обозначаемый символами "$1". Введем команду:

```
mc books
```

В результате выполнения макроса MC с формальным параметром BOOKS будет создан каталог BOOKS; далее командой CD BOOKS он будет сделан текущим. Используя макросы, можно создавать последовательность команд для перемещения одного или нескольких файлов, как это сделано ниже:

```
doskey mv=copy $1 $2 $t del $1
```

Для создания макроопределения для команды MEM, чтобы она делала паузу после показа каждого экрана, введите команду:

```
doskey mem=mem $* /P
```

Примечания:

Команда DOSKEY позволяет расширить возможности управления командной строки DOS с помощью дополнительных клавиш, значения которых приведены ниже:

- Up - (Стрелка вверх) - отмена команды MS-DOS, введенной пользователем перед ее исполнением;

- Down - (Стрелка вниз) - отмена команды MS-DOS, введенной пользователем, после того как она запущена на выполнение;

- PgUp - (Листание вверх) - отмена всех команд MS-DOS, ранее введенных в текущем сеансе работы;

- PgDn - (Листание вниз) - отмена последней введенной команды MS-DOS.

Когда программа DOSKEY находится в памяти, пользователю предоставляется возможность редактировать командную строку с помощью дополнительных клавиш, обозначения которых приведены далее:

- Left - (Стрелка влево) - движение курсора на один символ назад;

- Right - (Стрелка вправо) - движение курсора на один символ вперед;

- Ctrl+Left - движение курсора на одно слово назад;

- Ctrl+Right - движение курсора на одно слово вперед;

- Home - (Начало) - движение курсора в начало строки;

- End - (Конец) - движение курсора в конец строки;

- Esc - очистка командной строки;

- F1 - копирование одного символа из командной строки в буфер. Это символ находится в буфере для быстрого вызова последней команды по вводу этого символа;

- F2 - копирование из буфера всех символов, но не включая символа, введенного после нажатия клавиши F2;

- F3 - копирование содержимого буфера в командную строку. DOSKEY начинает копировать символы на позицию,указанную курсором в командной строке;

- F4 - удаление символов, начиная с символа, указанного текущим положением курсора, по символ, указанный вами. Если вам необходимо использовать эту клавишу для редактирования командной строки, нажмите сначала клавишу F4, а затем укажите символ, до которого необходимо уничтожить часть командной строки; DOSKEY удалит указанную часть командной строки, но сам введенный символ уничтожен не будет;

- F5 - копирование текущей команды в буфер и очистка командной строки, без выполнения команды;

- F6 - вывод в текущую позицию курсора символа "конец файла" (end-of-file (EOF)), (Ctrl+Z);

- F7 - вывод на дисплей списка всех команд, которые запомнила DOSKEY. Каждой введенной команде DOSKEY присваивает свой номер. Той команде, которая была введена позже всех, будет присвоен номер 1 и так далее для всех других команд;

- Alt+F7 - удаление из памяти всех запомненных команд;

- F8 - поиск в памяти команды, которую DOSKEY ранее запомнила. Для использования этой клавиши редактирования введите первый

символ или первые несколько символов команды, которую необходимо найти, и далее нажмите клавишу F8. DOSKEY выведет на дисплей найденную команду, которую далее можно редактировать в командной строке и запустить ее на выполнение;

- F9 - вывод на дисплей команды из памяти с ее номером по указанному ранее номеру. Для вывода на дисплей списка всех команд используйте клавишу F7.

- ALT+F10 - удаление всех макроопределений.

Если вы используете клавишу Insert для редактирования командной строки, то имеется возможность вводить текст между старым текстом, без его изменения. Однако, когда вы вы нажмете клавишу Enter для завершения ввода текста, клавиатура возвращается в режим замены текста. Для возвращения в режим вставки вновь нажмите клавишу Insert. Отменить режим вставки можно как с помощью клавиши Enter, так и с помощью ключа /OVERSTRIKE. Вы также можете изменить режим вставки текста, еще один раз нажав клавишу Insert.

Для создания макроопределений команд MS-DOS используются специальные символы команды DOSKEY. Значения специальных символов приведены ниже:

- $G или $g - перенаправление вывода. Эти символы, используемые DOSKEY, являются эквивалентом символу ">", который перенаправляет выходную информацию на устройство, отличное от стандартного устройства вывода - дисплея или в файл;

- GG или gg - добавление выходной информации в конец файла. Используйте эти символы для добавления выходной информации в указанный файл. Символы GG являются эквивалентом символов ">>";

- $L или $l - перенаправление ввода. Используйте эти символы для организации ввода информации с устройства, отличного от стандартного устройства ввода DOS - клавиатуры или с файла. Таким устройством, например, может быть последовательный порт COM1;

- $B или $b - направление выходной информации одной команды DOS на вход другой команды DOS. Аналогично использованию символа конвейеризации "|";

- $T или $t - разделение команд DOS между собой в строке макроопределения;

- $$ - указание в строке макроса знака доллара;

- $1... $9 - задание формальных параметров. Специальные символы $1... $9 используются для указания в одной строке макроопределения до 10 формальных параметров. Формальные параметры могут в даль-

нейшем использоваться при вызове макросов DOSKEY. Использование символов $1... $9 в команде DOSKEY аналогично использованию символов $1... $9 для задания списка формальных параметров в командных файлах;

- $* - завершение строки макроопределения для всех ранее введенных в макрос команд DOS.

Удаление созданного макроопределения производится командой:

doskey имя_макроса =

где параметр *имя-макроса* задает имя ранее созданного макроопределения.

DOSSHELL

Запуск графического интерфейса (оболочки) MS-DOS. Команда внешняя. Файл DOSSHELL.EXE. Оболочка MS-DOS Shell призвана облегчить пользователю процесс общения с компьютером посредством упрощения выполнения некоторых операций и более понятного представления информации, чем командная строка DOS.

Оболочка позволяет осуществлять управление собой как с помощью клавиатуры, так и манипулятором типа "мышь". После загрузки программы в память экран компьютера примет вид, который представлен на рисунке 8.3

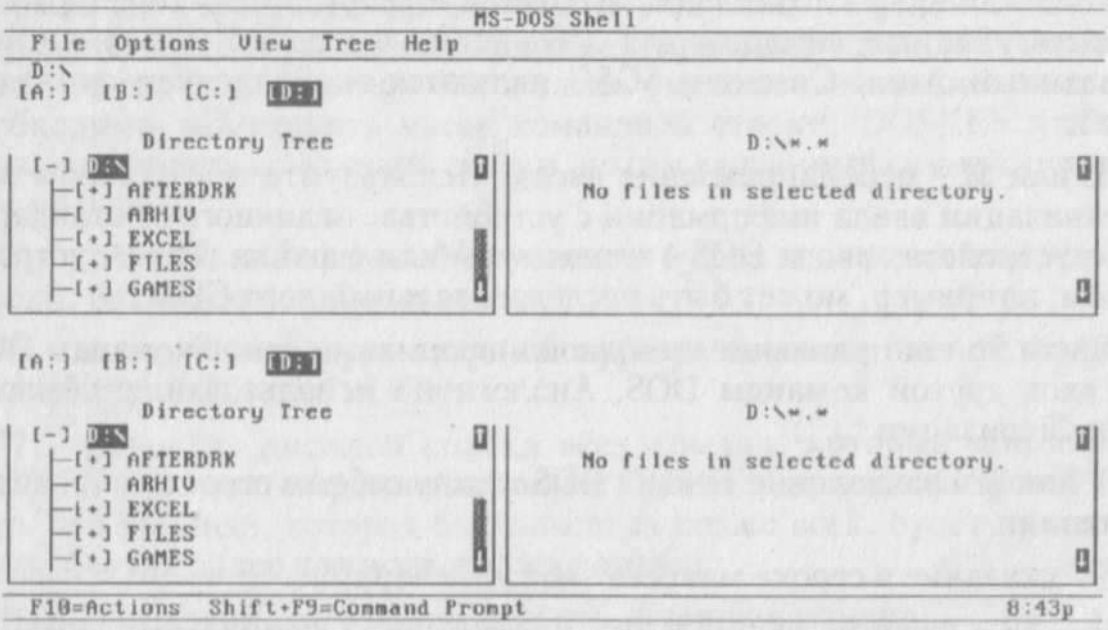


Рис. 8.3

Условно экран можно разделить на следующие части:

- Строка главного меню, содержащая в себе следующие пункты: "File", "Options", "View", "Tree", "Help", "MS-DOS Shell".
- Рабочая часть экрана, содержащая в зависимости от режима структуру каталогов и файлов на выбранном ранее диске. Выбор диска осуществляется с помощью клавиш управления курсором в строке между главным меню и рабочей областью.
- Строка подсказки, расположенная внизу. В строке подсказки отображается также и системное время.

Далее мы приводим описание основных функций, доступных через главное меню. Вход в главное меню осуществляется с помощью клавиши Alt и далее либо с помощью клавиш управления курсором и Enter, либо с помощью нажатия клавиш, обозначающих первую букву пунктов меню. Для перехода между различными частями экрана используется клавиша Tab.

Пункт главного меню "File" содержит:

- "Open" - открытие файла для дальнейшей работы с ним;
- "Run..." - запуск программ на выполнение;
- "Print" - печать содержимого ранее выбранного файла на принтере;
- "Associate..." - организация связи одного файла с другим;
- "Search" - поиск указанного файла на диске;
- "View File Contents" - просмотр содержимого указанного файла;
- "Move F7" - перемещение одного или группы файлов;
- "Copy" - копирование одного или нескольких файлов;
- "Delete" - удаление файла (ов);
- "Rename" - переименование одного или группы файлов;
- "Change Attributes" - изменение атрибутов у указанного файла или группы файлов;
- "Create Directory" - создание каталога;
- "Select All" - пометить все файлы;
- "Deselect All" - снять пометку со всех файлов.

Пункт главного меню "Options" содержит:

- "Confirmation" - этот пункт меню позволяет задать режим, когда MS-DOS Shell будет автоматически выдавать DOS подтверждение при следующих операциях:
  - удалении файлов (Confirm on Delete);
  - замене файлов (Confirm on Replace);
  - работе с "мышью" (Confirm on Mouse Operation);
- "File Display Options" - в этом пункте меню, вы можете управлять порядком сортировки файлов и типом файлов, выводимых на экран. Сортировка проводится:
  - по имени ("Name");
  - по расширению ("Extension");
  - по дате и времени создания ("Date");
  - по размеру ("Size");
  - по порядку записи файлов на диск ("Disk Order").

  Кроме вышеперечисленных вариантов сортировки, можно указать режим отображения "скрытых" и "системных" файлов, а также режим вывода списка файлов без учета порядка их записи на диск;
- "Select Across Directories" - выбор текущего каталога;
- "Show Information" - вывод на дисплей дополнительной информации о диске;

- "Show Information" - вывод на дисплей дополнительной информации о диске;
- "Enable Disk Swapper" - разрешение использования обмена информацией между дисками компьютера;
- "Display" - выбор режимов работы дисплея;
- "Colors" - задание цветовой палитры при работе с MS-DOS Shell.

Пункт главного меню "View" содержит:

- "Single File List" - изменение вида представления информации на экране. Справа отображается список каталогов текущего диска и текущий каталог (помечен цветом), а справа список файлов текущего каталога. Перемещение по каталогам и файлам осуществляется с помощью клавиш управления курсором. Выбранный файл также, как и выбранный каталог помечается цветом;
- "Dual File List" - в рабочем поле экрана организуются две области, соответствующие двум выбранным дискам. Вид окна для каждого диска аналогичен виду окна для режима "Single File List". Переключение между окнами осуществляется с помощью клавиши Tab;
- "All files" - при выборе этого пункта экран, содержащий рабочую информацию, выглядит следующим образом:
  - слева отображается информация о текущем каталоге текущего диска и самом диске;
  - справа выводится общий список файлов текущего диска;
- "Programm /File List" - в этом режиме работы экран разбивается на 3 части, две из которых расположены справа и слева от центра экрана, а третья внизу. Та часть экрана, которая расположена слева от центра, содержит информацию о структуре каталогов текущего диска, напротив нее выводится список файлов текущего каталога; внизу расположено окно с именем "Main", войдя в которое, можно выбрать следующие пункты меню окна "Main":
  - "Command Prompt" - выход в командную строку DOS;
  - "Editor" - запуск текстового редактора MS-DOS Editor;
  - "MS-DOS QBasic" - запуск среды программирования MS-DOS QBasic;
  - "[Disk Utilites]" - выход в подменю "Disk Utilites" для работы с файловой системой из среды MS-DOS Shell. В подменю "Disk Utilites" доступны следующие команды:
  - "Disk Copy" - копирование гибких дисков;
  - "MS Anti-Virus" - проверка вашего компьютера на вирусы с помощью утилиты MSAV;
  - "MS Backup" - проведение резервного копирования информации утилитой MSBACKUP;
  - "Quick Format" - быстрое форматирование дисков;
  - "Undelete" - восстановление случайно стертых пользователем файлов;
  - "Repaint Screen Shift+F5" - обновление экрана DOS-Shell до предыдущего экрана.

Пункт главного меню "Tree" позволяет просмотреть структуру дерева каталогов на указанных дисках. Список подпунктов:

- "Expand One Level +" - разворачивание структуры дерева каталогов до 1 каталога;
- "Expand Branch *" - увеличение числа видимых на экране каталогов;
- "Expand All Ctrl + *" - показывать максимально большее количество каталогов, одновременно с показом списка файлов в текущем каталоге;
- "Collapse Branch" - свернуть показ дерева каталогов.

Пункт главного меню "Help" содержит подробную справочную информацию о работе MS-DOS Shell, ее управлении, и тонкостях, имеющих место при задании команд в этой оболочке MS-DOS. Список подменю:

- "Index" - поиск по темам;
- "Keyboard" - управление MS-DOS Shell с помощью клавиатуры;

- "Shell Basics " - в этом разделе раскрываются основные принципы функционирования MS-DOS Shell;
- "Commands" - раздел содержит сведения о командах MS-DOS Shell;
- "Procedures" - в этом разделе раскрываются способы управления оболочкой;
- "Using Help" - раздел содержит сведения, связанные с использованием как самого пункта главного меню "Help", так и с получением помощи в других меню оболочки;
- "About Shell" - этот пункт меню содержит краткую информацию о версии MS-DOS Shell и авторских правах на эту оболочку фирмы Microsoft. Кроме управления MS-DOS Shell с помощью клавиатуры и мыши, вы можете осуществить ограниченное управление оболочкой с помощью командной строки DOS.

Формат команды:

В текстовом режиме:

dosshell /T:*разр n* /B;

В графическом режиме:

dosshell /G:*разр n* /B

Параметры:

разр - указание типа режима разрешения дисплея. Допускается использование символов L, M и H для указания соответственно режимов низкого, среднего и высокого разрешения. По умолчанию принимается значение разрешения максимальное для данного типа дисплея.

n - указание числа линий в установленном режиме разрешения. Посмотреть допустимые значения n для каждого из режимов работы дисплея можно в пункте главного меню "Options" в подменю "Display".

Ключи

/T	Запуск MS-DOS Shell в текстовом режиме;
/B	Запуск MS-DOS Shell в черно-белом режиме;
/G	Запуск MS-DOS Shell в графическом режиме;

Пример использования:

Для старта MS-DOS Shell в графическом режиме введите команду:

dosshell /G

Примечания:

- Находясь В MS-DOS Shell, нельзя осуществить запуск Microsoft Windows. Если же вам необходимо совместно использовать MS-DOS Shell и Microsoft Windows, запустите сначала Windows, а затем из среды Windows оболочку MS-DOS Shell.
- Все текущие установки MS-DOS Shell: для программ, тем и групп, разрешения экрана и цветовой палитры и т. д. записываются в

файле DOSSHELL.INI. Файл DOSSHELL.INI создается при выходе из оболочки или изменении программного меню и обязательно должен располагаться на незащищенном от записи диске, так как он постоянно модифицируется. Местоположение файла DOSSHELL.INI можно указать в файле AUTOEXEC.BAT для ввода этого параметра в окружение MS-DOS Shell, например:

```
dosshell = c:\dos
```

В результате выполнения этой команды, DOSSHELL передается информация о том, что файл DOSSHELL.INI находится на диске C: в каталоге \DOS. Если файл DOSSHELL.INI будет удален или перемещен в другое место, то при очередном запуске DOSSHELL сама создаст его.

- Для ускорения работы DOSSHELL вы можете также указать местоположение временных файлов, сообщив этот параметр переменной TEMP в файле AUTOEXEC.BAT.
- При работе с оболочкой DOSSHELL она подсвечивает доступные пункты меню. Те пункты меню, которые нельзя выбрать, подсвечены неярким цветом. Пункты меню можно быстро выбрать при помощи клавиш, подсвеченных другим цветом, чем остальная часть текста.

(!) EDIT

Запуск текстового редактора MS-DOS Editor, для создания и изменения текстовых и ASCII файлов. MS-DOS Editor является полноэкранным текстовым редактором, работая в котором вы можете создавать, редактировать, записывать и распечатывать текстовые файлы в формате кодов ASCII. Работа с MS-DOS Editor осуществляется при помощи окон и диалоговых окон. Команда внешняя. Файл EDIT.COM. После запуска программы экран имеет вид, представленный на рисунке 8.4.

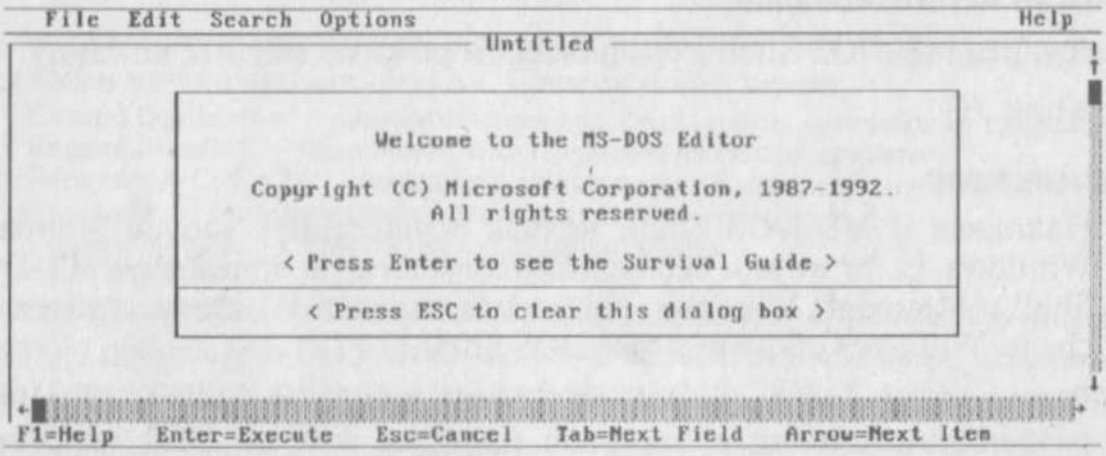


Рис. 8.4

Условно экран MS-DOS Editor можно разделить на 3 части:

I область меню, где выводится список основных пунктов с указанием имени текущего рабочего файла;

II рабочее поле, содержащее редактируемый текст;

II область подсказки, которая содержит сведения об использовании основных команд MS-DOS Editor.

После загрузки программы в память на экране появится диалоговое окно, в котором пользователю предлагается либо сразу перейти к работе с файлами, либо войти в справочную систему MS-DOS Editor. Нажав клавишу Enter, вы перейдете в систему помощи MS-DOS Editor, о которой будет рассказано позже, а, нажав клавишу Esc, очистите появившееся диалоговое окно и приступите непосредственно к работе с текстами.

Область меню редактора MS-DOS Editor содержит в себе следующие пункты: "File", "Edit", "Search", "Options", "Help", используя которые, оператор имеет возможность управлять работой редактора.

Вход в главное меню программы осуществляется нажатием клавиши Alt. Описание содержащихся в каждом из пунктов главного меню подпунктов приведено ниже.

Пункт главного меню "FILE" содержит следующие подпункты:

- "New" - создание нового текстового файла;
- "Open..." - открытие файла и его загрузка в оперативную память для дальнейшей работы с ним. По умолчанию принимается, что загружаемый файл имеет расширение .TXT;
- "Save" - запись файла с тем же именем, что и при его загрузке на диск. Если записываемый файл имени не имеет, то DOS попросит его ввести;
- "Save as..." - запись файла с измененным именем и расширением;
- "Print..." - распечатка текста или его фрагмента на принтере. После выбора этого пункта откроется диалоговое окно, в котором необходимо выбрать, что вы хотите распечатать:
  - помеченный фрагмент текста ("Selected Text Only");
  - весь текст целиком ("Complete Document");
- "Exit" - выход из среды редактора обратно в MS-DOS.

В пункте главного меню "Edit" содержит пункты:

- "Cut Shift-Del" - разрезать текст на части;
- "Copy Ctrl-Ins" - копирование фрагмента текста;
- "Paste Shift-Ins" - склеивание фрагментов текста;
- "Clear Del" - очистка диалогового окна от текста.

Пункт меню "Search" содержит пункты:

- "Find..." - задание поиска в тексте указанной строки. При выборе этого пункта меню открывается еще одно подменю, в котором пользователь может задать следующие установки:
  - "[ ] Match Upper/Lowercase" - игнорировать или нет прописные и строчные буквы;

- "[ ] Whole Word" - искать целое слово или нет;
- "Repeat Last Find" - повторить поиск последней указанной строки. Раскрывающееся при выборе этого пункта подменю аналогично подменю пункта "Find...";
- "Change..." - изменение указанной строки поиска на другую. Раскрывающееся подменю позволяет заменить введенную ранее строку поиска на другую и установить режимы проверки прописных и строчных букв, а также целых слов. Находящиеся внизу этого подменю указатели позволяют:

"Find and Verify" - осуществить поиск и проверку указанных строк;
"Change All" - изменить все строки поиска;
"Cancel" - закончить работу с этим окном;
"Help" - получить помощь по работе с этим пунктом подменю. Хочется отметить, что редактор MS-DOS Editor позволяет пользователю получать помощь практически в любых режимах работы выбором указателя "Help" в нижней части окон, либо с помощью нажатия клавиши "F1".

Пункт главного меню "Options" содержит:

- "Display..." - установка цветов фона и символов, а также режима вывода нижней строки подсказки "Scroll Bars" и символов табуляции;
- "Help Path" - задание местоположения на диске файла EDIT.HLP, содержащего справочную информацию о программе MS-DOS Editor. Если этот параметр опущен, то предполагается, что файл EDIT.HLP размещен в том же каталоге, что и файл EDIT.COM.

Пункт "Help" главного меню содержит:

"Getting Started" - запуск справочной системы MS-DOS Editor.

После выбора этого пункта раскроется окно, содержащее строку выбора темы справки, в которой пользователь видит следующие пункты: "Getting Started", "Keyboard", "Back". Область, содержащая справочную информацию по темам:

- "Using Help" - информация об использовании справочной службы MS-DOS Editor;
- "Using Menus & Commands" - использование "мыши" и клавиатуры для управления средой текстового редактора;
- "Using Dialog Box" - информация об управлении диалоговыми окнами с помощью клавиатуры и "мыши";
- "MS-DOS Editor Options" - раздел содержит описание запуска MS-DOS Editor из командной строки DOS. Этот раздел подробно будет обсужден далее;
- "Copyright & TradeMarks" - в этом разделе приводится информация об авторских правах на программу и условиях ее распространения;

Нижняя часть окна представляет собой строку подсказки с указанием основных клавиш управления справочной системой. Верхняя часть экрана используется для перехода между темами справки.

Подменю "Keyboard" пункта "Help" позволяет посмотреть информацию об управлении средой MS-DOS Editor с помощью клавиатуры:

- "Shortcut Keys" - описание основных "горячих" клавиш управления редактором;
- "Help Keys" - описание клавиш для вызова и управления справочной системой;
- "Cursor - Movement Keys" - описание клавиш управления положением курсора;
- "Text - Scrolling Text" - информация по клавишам, управляющим перемещением текста;
- "Text - Selection Keys" - справка по клавишам выделения текста;
- "Insert and Copy" - описание клавиш выделения и копирования текста;

- "Delete Keys" - описание клавиш выделения текста;
- "Find and Change Keys" - информация по клавишам поиска и изменения текста.

Войти в раздел справочной системы "Keyboard" можно как при помощи верхней строки справочной системы, так и выбрав раздел "Keyboard" в пункте "Help" главного меню.

Пункт "Back" в верхней части экрана - возврат в предыдущее окно справочной системы из ранее выбранного.

Кроме вышеперечисленных пунктов, в подменю "Help" входит еще пункт "About", который содержит общую информацию о программе MS-DOS Editor.

Управление средой редактора можно осуществлять также и из командной строки DOS с помощью специальных параметров и ключей. Их описание мы приводим далее.

Формат команды:

edit *диск:\маршрут\имя_файла* /B /G /H /NOHI

Параметры:

диск:\маршрут\имя_файла - указание полного идентификатора текстового файла. Этот файл будет открыт и загружен в память MS-DOS Editor при его запуске, если он существует. Если же его не существует, MS-DOS Editor автоматически создаст его.

Ключи

/B	установка черно-белого режима отображения. Используйте этот ключ, когда изображение выглядит некорректно на монохромном мониторе;
/G	устранение эффекта "снег" для CGA-монитора;
/H	установка максимального числа линий изображения для данного типа монитора;
/NOHI	разрешение использования для отображения информации только 8 основных цветов (с 0 по 7). Обычно MS-DOS использует для вывода изображения все 16 основных цветов.

Предупреждение: MS-DOS Editor не будет работать, если файл QBASIC.EXE не находится в текущем каталоге или к нему не указаны пути доступа или если он не находится в одном каталоге с файлом EDIT.COM. Если вы удалите файл QBASIC.EXE с вашего жесткого диска, вы не сможете работать с MS-DOS Editor.

Примечания:

- При работе с редактором MS-DOS Editor он подсвечивает доступные пункты меню. Те пункты меню, которые нельзя выбрать, подсвече-

ны неярким цветом. Пункты меню можно быстро выбрать при помощи "горячих" клавиш; они подсвечены другим цветом, чем остальная часть текста.

- Некоторые типы мониторов не поддерживают подсвечивание "горячих" клавиш. Если ваш монитор не подсвечивает "горячую" клавишу, используйте в этом случае ключ /B (для CGA мониторов) и ключ /NOHI для мониторов, не поддерживающих отображение утолщенных символов.

  При работе с Windows драйвер EMM386 организует поддержку вашей системой расширенной памяти и позволяет также рассматривать область старшей памяти как расширенную память. Для получения более подробных сведений о драйвере EMM386.EXE, обратитесь в главу "Загружаемые драйверы устройств".

Формат команды:

emm386 ON|OFF|AUTO W=ON|OFF

Для вывода на дисплей текущих установок драйвера EMM386 просто введите:

emm386

Параметры:

ON|OFF|AUTO - активизация драйвера EMM386 (ключ установлен в положение N), или запрещение поддержки EMS (ключ находится в положении OFF), или включение автоматического режима работы (ключ в положении AUTO). В автоматическом режиме любая программа, требующая доступа к EMS, сразу получит его. По умолчанию рассматриваемый ключ установлен в положение ON.

W=ON|OFF - разрешение поддержки сопроцессора Weitek, если ключ /W установлен в положение ON, и запрещение поддержки сопроцессора Weitek, если ключ /W установлен в положение OFF. По умолчанию ключ /W находится в положении OFF.

Примечания:

- Для того, чтобы использовать команду EMM386, ваша система должна иметь процессор 80386 и выше, а также драйвер EMM386, подключенной к ОС директивой DEVICE файла CONFIG.SYS. В том случае, если вы попытаетесь использовать команду EMM386 на компьютере, не имеющим процессора 80386 и выше, а также с ними совместимых, на дисплее появится следующее сообщение:

  EMM386 driver not installed

- Для организации поддержки расширенной памяти в случае, когда драйвер EMM386 не был подключен DOS в файле CONFIG.SYS, можно обеспечить поддержку расширенной памяти, загрузив EMM386 в память с ключом ON.
- Когда с помощью ключа OFF отключается поддержка EMM386 расширенной памяти, заголовок драйвера EMM386.EXE изменяет программы, которые не могут получить доступа к расширенной памяти. Этот прием применяется для запуска программ, не являющихся VCPI-приложениями для работы в стандартном режиме Windows 3.0.
- В том случае, когда ключ /W установлен в положение ON EMM386 разрешает поддержку сопроцессора Weitek. Однако следует помнить, что для нормальной работы сопроцессора ему также должна быть доступна область старшей памяти (HMA). Для того, чтобы обеспечить поддержку старшей памяти, используйте директиву DOS=HIGH в файле CONFIG.SYS.

  При попытке использования ключей W = ON или W = OFF на компьютере, не имеющем сопроцессор Weitek, MS-DOS выведет на дисплей следующее сообщение:

  Weitek Coprocessor not installed

ERASE

Удаление одного или группы файлов. Команда внутренняя. Команда ERASE представляет собой другое обозначение команды DEL.

EXIT

Выход из вторичного командного процессора MS-DOS (COMMAND.COM) и возврат управления первичному командному процессору, если он загружен в память. Команда внутренняя.

Формат команды:

exit

Примечания:

При загрузке командного процессора в память постоянно (ключ /P в команде COMMAND) из него нельзя выйти командой EXIT. Дополнительную информацию по команде COMMAND можно получить в описании этой команды.

EXPAND

Разворачивание файлов DOS в процессе инсталляции (установки) DOS с установочных на ваш компьютер. Команда применяется для работы только с файлами, находящимися на установочных дискетах. Команда внешняя. Файл EXPAND.EXE.

Формат команды:

expand *диск1:\маршрут1\имя_файла1 диск2:\маршрут2\имя_файла2*

Если команда EXPAND загружается на выполнение с установочного диска, необходимо указывать полный идентификатор файла, который необходимо преобразовать. В этом случае формат команды имеет вид:

expand *диск2:\маршрут2\имя_файла2*

В том случае, если вам необходимо, чтобы EXPAND сама запросила местонахождение "сжатого" файла и полный идентификатор развернутого файла, команда вводится в следующем виде:

expand

Параметры:

диск1\маршрут1\имя_файла1 - указание местоположения "сжатого" файла, подлежащего преобразованию в нормальный вид. Для указания местоположения файла нельзя использовать групповые шаблоны * и ?.

диск2\маршрут2\имя_файла2 - указание местоположения или развернутого файла(ов) после выполнения команды EXPAND. Эти параметры включают в себя указание диска, каталога и имени файла, или комбинацию этих параметров.

Примеры использования:

Допустим, вы удалили файл SORT.EXE, находившийся в каталоге DOS на диске C:. Для его восстановления можно использовать несколько способов.

Один из них заключается в преобразовании файла SORT.EX_, хранящегося на установочных дисках. Вообще говоря, для установочных дискет более часто применяется обозначение "дистрибутивные" дискеты, но будет использовать понятие "установочная" дискета, как лучше отражающее смысл дистрибутивных дискет. Перед восстановлением файла SORT.EX_ просмотрите файл PACKING.LST на дистрибутивном диске 1. Далее введите следующую команду:

```
expand a:\sort.ex_ c:\dos\sort.exe
```

Команда EXPAND выполнит развертывание и копирование файла SORT.EX_ и преобразует этот файл в нормальный файл SORT.EXE, расположенный на диске C в каталоге \DOS. Если же вы хотите работать с EXPAND в диалоговом режиме, задайте команду EXPAND в следующем виде:

```
expand
```

После ввода команды на экране появится сообщение:

```
Type the location and name of the compressed file
you want to expand.
(Введите местоположение и(или) имя "сжатого" файла для его преобразования).
```

Для разворачивания и копирования файла SORT.EX_ с установочного диска A:, введем:

```
a:\sort.ex_
```

Далее EXPAND выведет следующее сообщение:

```
Type the location and/or name you want to give the expanded file.
(Введите местоположение и(или) имя преобразованного файла)
Expanded file:
```

Введем на запрос системы полный идентификатор преобразованного файла:

```
c:\dos\sort.exe
```

Примечания:

"Сжатые" файлы ОС, которые хранятся на установочных дисках, имеют расширение, оканчивающееся символом "_". Когда вы инсталлируете MS-DOS, она сама разворачивает "сжатые" файлы и осуществляет их копирование на ваш системный диск. Однако в случае их затирания или порчи одним из путей их восстановления как раз и является использование команды EXPAND. О другом пути восстановления стертых файлов мы расскажем в описании утилиты UNDELETE.

(!) FASTHELP

Вывод на дисплей списка всех команд MS-DOS 6.0 и получение краткой справки по каждой команде. Для получения более детальной информации по интересующей команде пользователю следует использовать команду Help. Команда внешняя. Файл FASTHELP.EXE.

Формат команды:

fasthelp *команда*

Другим вариантом использования команды является следующий:

команда /?

Параметр:

команда - указание команды MS-DOS, по которой необходимо получить справочную информацию. Если команда вводится без параметра, то FASTHELP выводит на дисплей список всех команд MS-DOS 6.0.

Примеры использования:

Для получения справочной информации о команде, например XCOPY, введите:

```
fasthelp xcopy
```

либо используйте другой вариант этой же команды:

```
xcopy /?
```

Примечания:

В приведенных примерах использования FASTHELP приведены оба метода использования FASTHELP. Однако следует отметить, что в последнем варианте FASTHELP работает быстрее.

FASTOPEN

Ускорение открытия DOS файлов для повышения скорости работы вашего компьютера с большими каталогами. Команду нельзя запускать на выполнение, находясь в среде Microsoft Windows. FASTOPEN сохраняет в памяти информацию об однажды открытых файлах, что при дальнейшей работе с ними уменьшает время их открытия. Команда внешняя. Файл FASTOPEN.EXE.

Формат команды:

fastopen *диск1: = n(1) дискi: = n(i)* /X

Для загрузки FASTOPEN в процессе обработки ОС файла CONFIG.SYS используйте следующий формат команды:

install = *диск:\маршрут***fastopen.exe** *диск1: = n(1) дискi: = n(i)* /X

Предупреждение: для сохранности ваших данных не используйте программы дефрагментации файлов, например DEFRAG, пока в памяти находится утилита FASTOPEN.

Параметры:

диск:\маршрут - указание полного пути доступа к файлу FASOPEN.EXE.

диск1: - указание жесткого диска, при работе с которым FASTOPEN будет запоминать расположение файлов.

дискi: - указание (i) жесткого диска, при работе с которым FASTOPEN будет запоминать расположение файлов.

n - указание числа записей о файлах, которые FASTOPEN будет запоминать одновременно. Значение n должно находится в пределах от 10 до 999. По умолчанию это значение равно 48.

Ключ	
/X	создание области памяти, хранящей информацию о файлах в расширенной памяти. Для использования этого ключа компьютер должен быть оснащен расширенной памятью, удовлетворяющей спецификации LIM EMS 4.0.

Примеры использования:

Допустим, вам необходимо хранить информацию о 100 файлах, расположенных на диске C:. Для решения этой проблемы добавьте в ваш файл CONFIG.SYS следующую строку:

```
install = c:\dos\fastopen.exe c:=100
```

или просто добавьте команду FASTOPEN C:=100 в файл AUTOEXEC.BAT. Эту же команду можно ввести в командной строке DOS.

Примечания:

- Команда FASTOPEN работает только с жесткими дисками, при условии того, что они не являются сетевыми.
- FASTOPEN не может одновременно работать с более чем 24-ю логическими дисками винчестера (винчестеров).
- При использовании FASTOPEN для работы с несколькими логическими дисками винчестера максимальное число параметров *n* не должно превышать 999 суммарно для всех дисков.
- Установки FASTOPEN, заданные один раз, далее не меняются в сеансе работы с компьютеров. Для изменения параметров FASTOPEN необходимо изменить их, а затем перезапустить компьютер.
- Нельзя запускать утилиту FASTOPEN из оболочки MS-DOS Shell, так это может привести к зависанию вашего компьютера.
- В процессе работы утилита FASTOPEN расходует приблизительно 48 байт памяти на каждую запись об имени файла.

FC

Сравнение двух файлов или двух групп файлов. Команда внешняя. Файл FC.EXE.

Формат команды:

Для выполнения сравнения файлов в формате кодов ASCII используйте следующий вариант задания команды FC:

fc /A /C /L /LBn /N /T /W /*nnnn диск1:\маршрут1\имя_файла1 диск2:\маршрут2\имя_файла2*

Для выполнения двоичного сравнения файлов, используйте следующий формат команды:

fc /B *диск1:\маршрут1\имя_файла1 диск2:\маршрут2\имя_файла2*

Параметры:

диск1:\маршрут1\имя_файла1 - указание полного идентификатора первого файла (группы файлов), выбранного(ных) для сравнения.

диск2:\маршрут2\имя_файла2 - указание полного идентификатора второго файла (группы файлов), выбранного(ных) для сравнения.

Ключи

/A	установка формата выходной информации команды FC в ASCII кодах. При использовании этого ключа FC выводит на дисплей только первую и последнюю строки участка, в которых обнаружены различия;
/C	при использовании этого ключа игнорируется тип букв (прописные или заглавные);
/L	сравнение файлов в режиме кодов ASCII. FC сравнивает две строки файлов и в случае обнаружения различий выводит их на экран. Этот режим по умолчанию установлен для всех файлов, не имеющих расширений .EXE, .COM, .SYS, .OBJ, .LIB или .BIN.;
/LBn	установка числа строк, хранящихся во внутреннем буфере. По умолчанию буфер хранит 100 строк. Если количество строк выходной информации FC превышает объем буфера, то сравнение двух файлов прекращается;
/N	вывод на дисплей номера строки, во время ASCII-сравнения;
/T	задание режима работы FC, при котором символы табуляции не будут преобразовываться в пробелы. По умолчанию символы табуляции установлены в каждую восьмую позицию строки;

/W установка режима сжатия свободного пространства (табуляций и пробелов) во время сравнения. Если строка во время сравнения содержит символы табуляции и пробелы, то при использовании этого ключа они будут преобразованы в один пробел. Следует отметить, что с ключом /W команда FC игнорирует и не сравнивает пробелы в начале и конце строки;

/*nnnn* указание числа строк, которые должны совпадать после выявленного командой FC различия. Если параметр *nnnn* больше числа обнаруженных совпадающих строк, то совпадающие строки рассматриваются как несовпадающие.

/B сравнение файлов производится в двоичном коде. Утилита FC сравнивает два файла байт за байтом с начала и до их конца. При обнаружении несовпадения счетчики байтов не будут равны. Этот ключ установлен по умолчанию для файлов с расширениями .EXE, .COM, .SYS, .OBJ, .LIB или .BIN.

Примеры использования:

Для сравнения двух тестовых файлов с именами MONTHLY.TXT и NOTES.TXT с выводом результата сравнения в кодах ASCII введите:

```
fc /A monthly.txt notes.txt
```

Для выполнения двоичного сравнения двух пакетных файлов START1.BAT и START2.BAT введите команду:

```
fc /B start1.bat start2.bat
```

В результате выполнения этой команды экран примет следующий вид:

```
00000002: 72 43
00000004: 65 3A
0000000E: 56 92
00000012: 6D 5C
00000013: 0D 7C
00000014: 0D 0A
00000015: 0A 0D
0000001E: 43 7A
0000001F: 09 0A
00000022: 72 44
...
...
...
000005E0: 00 61
000005E1: 00 73
000005E2: 00 73
```

```
000005E3: 00 69
000005E4: 00 67
000005E5: 00 6E
000005E6: 00 6D
000005E7: 00 65
000005E8: 00 6E
FC: START2.BAT longer than START1.BAT
(Файл START2.BAT длиннее START1.BAT)
```

Если же файлы START1.BAT и START2.BAT не имеют различий, FC выведет на дисплей сообщение следующего характера:

```
FC: no differences encountered
```

Для сравнения каждого файла с расширением .BAT с файлом NEWS.BAT введите команду:

```
fc *.bat news.bat
```

Для сравнения файла NEWS.BAT на диске C: с файлом NEWS.BAT на диске D: задайте команду FC в виде:

```
fc c:new.bat d:*.bat
```

Для сравнения всех файлов с расширением .BAT, которые расположены на диске C:, со всеми командными файлами на диске D: введите:

```
fc c:\*.bat d:\*.bat
```

Примечания:

- Когда вы используете команду FC для выполнения сравнения в кодах ASCII, при обнаружении различий между двумя файлами DOS выводит на дисплей имя первого из сравниваемых файлов, первый файл с первой и по последнюю строку, имя второго файла, строку первого файла, в которой обнаружены различия, далее следует имя второго сравниваемого файла, потом выводится содержимое второго файла с первой по последнюю строку, следующая строка содержит результаты сравнения обоих файлов.
- При использовании ключа /B для проведения двоичного сравнения файлов MS-DOS выводит информацию в следующем формате:

  ```
  xxxxxxxx: ii jj
  ```

  Значение "xxxxxxxx" указывает шестнадцатеричный адрес для пары сравниваемых байтов, начиная от начала файла. Этот адрес равен 00000000 при начале сравнения файлов. Шестнадцатеричные значения *ii* и *jj* определяют различные байты в сравниваемых файлах.

- При использовании групповых шаблонов * и ? в именах файлов работа команды FC имеет некоторые особенности. Если вы используете групповой шаблон при указании имени первого из сравниваемых файлов, утилита FC будет сравнивать все указанные файлы с файлом, указанном в параметре *имя_файла2*. Если групповой шаблон используется при указании параметра *имя_файла2*, то FC сравнивает файл, указанный в параметре *имя_файла1* со всеми файлами, указанными в параметре *имя_файла2*.
- При выполнении сравнения файлов в формате кодов ASCII FC использует внутренний буфер, размещаемый в памяти. В случае, если количество строк выходной информации превышает размер выходного буфера, работа FC прерывается, и на дисплей выводится сообщение:

  Resynch failed. Files are too different.
  (Сравнение прекращено. Файлов слишком много)

- При сравнении файлов в двоичных кодах сравнение файлов, размер которых превышает объем оперативной памяти, производится порциями. В этом случае FC выводит информацию о несовпадающих байтах не во всем файле, а только в его части, загруженной в память.

FDISK

Разбиение жесткого диска на разделы. Команда внешняя. Файл FDISK.EXE. После запуска утилиты на экране появится меню, общий вид которого приведен на рисунке 8.5.

```
                    MS-DOS Version 6
                Fixed Disk Setup Program
           (C)Copyright Microsoft Corp. 1983 - 1993

                      FDISK Options

Current fixed disk drive: 1

Choose one of the following:

1. Create DOS partition or Logical DOS Drive
2. Set active partition
3. Delete partition or Logical DOS Drive
4. Display partition information


Enter choice: [1]




Press Esc to exit FDISK
```

Рис. 8.5

Команда выполняет разбиение диска на разделы (от 1 до 4) для размещения как операционной системы MS-DOS, так и других ОС. Для работы с ОС MS-DOS обычно создается два раздела:

- первичный раздел, который используется для загрузки операционной системы;
- расширенный раздел, в котором организуются отдельные тома, действующие как логические диски. В расширенном разделе можно создавать тома с обозначениями от D: до Z:. Величина разделов, как первичного, так и вторичного может быть практически любым.

Разбиение жесткого диска происходит в диалоговом режиме, с помощью системы меню. Меню состоит из следующих пунктов:

1. Create DOS partition or Logical DOS Drive (Создание раздела DOS или логического диска DOS).
2. Set Active partition (Установка активного раздела).
3. Delete partition or Logical DOS Drive (Удаление раздела или логического диска DOS).
4. Display partition Information (Вывод на дисплей информации о разделах).

При наличии в машине второго жесткого диска на дисплее будет отображен еще один пункт меню:

5. Select Next Fixed Drive (Выбор следующего жесткого диска).

Вход в пункты меню осуществляется набором цифры, соответствующей выбранному пункту меню в указанном поле ввода.

При выборе 1 пункта главного меню FDISK перед пользователем появится новое меню, в котором имеются следующие варианты:

1. Create Primary DOS Partition (Создание первичного раздела DOS).
2. Create Extended DOS Partition (Создание расширенного раздела DOS).
3. Create Logical DOS Drive(s) in the Extended DOS Partition (создание логических дисков DOS в расширенном разделе DOS).

Пункт 2 позволяет установить активный раздел, то есть тот раздел, с которого будет осуществляться загрузка ОС. Активным разделом для DOS следует назначать первичный раздел.

В пункте 3 главного меню FDISK вы можете удалить разделы или логические диски DOS.

При выборе этого пункта открывается меню, в котором имеются следующие пункты:

1. Delete Primary DOS Partition (Создание первичного раздела DOS).
2. Delete Extended DOS Partition (Создание расширенного раздела DOS).
3. Delete Logical DOS Drive(s) in the Extended DOS Partition (создание логических дисков DOS в расширенном разделе DOS).
4. Delete non-DOS Partition (Удаление не-DOS раздела).

Пункт 4 главного меню FDISK дает информацию об имеющихся разделах на жестком диске и логических дисках, на них организованных.

Пункт 5 главного меню дает возможность выбрать для работы другой жесткий диск.

Формат команды:

fdisk

Для вывода информации о логических дисках винчестера без входа в саму программу FDISK используйте команду FDISK в следующем формате:

fdisk /STATUS

Ключ

/STATUS	вывод информации о логических дисках винчестера без запуска утилиты FDISK.

Примечания:

Предупреждение: Не экспериментируйте с FDISK. Удаление диска или раздела может привести к полной потере данных на вашем жестком диске.

- Для изменения размера раздела вы должны сначала удалить старый раздел, а затем создать новый раздел необходимого вам размера.
- Максимальный размер раздела может составлять 2 гигабайта (GB).
- Команда FDISK не работает с дисками, созданными при помощи команды SUBST.
- FDISK не работает с сетевыми дисками или дисками, работающими под управлением программы INTERLNK. FDISK работает только с жесткими дисками, физически установленными на компьютере.
- Если ваш компьютер имеет "сжатые" диски, то FDISK не выводит на дисплей информацию о таких дисках. "Сжатый" диск создается на физическом диске в виде "скрытого" (только для чтения) файла "сжатого" тома (CVF). Утилита DBLSPACE присваивает CVF букву логического диска. Для получения подробной информации о

"сжатых" дисках посмотрите описание команды DBLSPACE, модуля DBLSPACE.BIN в главе "Состав DOS и ее основные модули" и описание драйвера DBLSPACE.SYS в главе "Загружаемые драйверы устройств".

FIND

Поиск указанной строки текста в файле или группе файлов. После окончания поиска строки в указанных файлах FIND выводит на дисплей все строки текста, совпадающие с указанной строкой. Команда внешняя. Файл FIND.EXE.

Формат команды:

find /V /C /N /I *строка диск:\маршрут\имя_файла*

Параметры:

строка - указание группы символов, которые необходимо найти. Строка текста, которую необходимо найти, должна быть заключена в двойные кавычки.

диск:\маршрут\имя_файла - указание местоположения файла (группы файлов), в которых будет производиться поиск указанной строки.

Ключи

/V	на дисплей будут выводиться все строки, не содержащие указанной строки поиска;
/C	вывод количества строк проверяемого файла, содержащих указанную фразу;
/N	вывод для каждой найденной линии текста ее номера;
/I	при использовании этого ключа команда FIND не различает прописные и заглавные буквы.

Примеры использования:

Для вывода на дисплей всех строк файла PRIMER.TXT, содержащих фразу "Good Year !", введите команду:

```
find "Good Year !" primer.txt
```

Для поиска строки, включающей в себя текст, заключенный в двойные кавычки, вы должны заключить текст в кавычках в еще одни двойные кавычки, как это сделано в примере, приведенном ниже:

```
find "Утилита DBLSPACE создает ""сжатые"" файлы." report.txt
```

Утилита FIND может быть также использована при организации командных файлов с целью поиска указанной строки в большом количестве файлов.

Очень часто команда FIND совместно используется с командой FOR пакетных файлов. Допустим, необходимо просмотреть все командные файлы текущего каталога и найти те из них, которые включают в себя команду ECHO OFF. Для организации процесса поиска указанных файлов создается командный файл, в состав которого входит команда:

```
for %f in (*.bat) do find "echo off" %f
```

Другой областью применения команды FIND является ее использование в качестве конвейера для других команд DOS. В качестве иллюстрации этого варианта использования команды FIND можно привести следующий пример. Необходимо осуществить на диске C: и вывести на дисплей список всех файлов, включающих строку "DOS".

Для решения этой проблемы введем команду:

```
dir c:\ /S /B | find "dos"
```

Для того, чтобы утилита FIND работала быстрее в операциях конвейеризации, установите местоположение переменной окружения TEMP в вашем файле CONFIG.SYS. Если вам необходимо, чтобы FIND не реагировала на верхний или нижний регистр букв при работе с выходной информацией команды DIR, используйте в команде FIND ключ /I.

Примечания:

- Если в команде FIND параметр *имя_файла* опущен, то она работает как фильтр для информации, вводимой со стандартного устройства ввода DOS (клавиатуры) и как конвейер в операциях перенаправления при использовании символа "|".
- При использовании ключей /V или /N вместе с ключом /C утилита FIND выводит на дисплей количество линий, в которых не найдена указанная фраза. При совместном использовании ключей /C и /N ключ /N игнорируется.
- Команда FIND не распознает символов перевода каретки (CR). Когда вы используете команду FIND для поиска текста в файле, включающем в себя символы CR, вы должны использовать строку, не имеющую символов CR. Для примера, FIND не найдет строку "report string", если между словами "report" и "string" не имеется символа пробела.
- По окончании своей работы FIND возвращает в DOS код завершения для дальнейшего использования его в прикладных программах или командных файлах.

Коды завершения FIND приведены ниже:

0	поиск прошел успешно, и найдена как минимум одна строка;
1	поиск прошел успешно и строк не найдено;
2	поиск строки не завершился успешно. В этом случае ошибка возникла во время поиска, и FIND не смог выдать отчет о результатах поиска.

FORMAT

Форматирование дисков для дальнейшего использования их операционной системой MS-DOS. Команда FORMAT.COM создает новый корневой каталог и таблицу размещения файлов на форматируемом диске. В процессе проверки также осуществляется проверка диска на дефекты, и в случае их обнаружения они исключаются из доступного дискового пространства. Если вы хотите использовать новый диск для работы с MS-DOS, то эта команда должна задана первой для форматирования диска. Команда внешняя. Файл FORMAT.COM.

Форматы команды:

format *диск:* /V:*метка* /Q /U /F:*размер* /B|/S
format *диск:* /V:*метка* /Q /U /T:*дорожек* /N:*секторов* /B|/S
format *диск:* /V:*метка* /Q /U /1 /4 /B|/S
format *диск:* /Q /U /1 /4 /8 /B|/S

Предупреждение: Команда FORMAT без специальных программных средств не позволяет форматировать диски на емкость, превышающую конструктивную емкость накопителя. К специальным программам, позволяющим обойти указанное ограничение команды FORMAT, относятся известные драйверы 800.COM, 900.COM, FDREAD.COM.

Параметры:

диск: - указание буквы, определяющей диск, который вы хотите отформатировать. Параметр является обязательным. Если в команде FORMAT вы используете параметр *диск:* без ключей, то указанный диск по умолчанию форматируется на максимальную емкость накопителя, которая перед форматированием определяется утилитой FORMAT.

В том случае, если диск уже был ранее отформатирован и вновь используете команду FORMAT без ключа /U, команда FORMAT уничтожает только старые корневой каталог и таблицу размещения файлов. Восстановить частично или полностью утраченную информацию на диске, отформатированном командой FORMAT без ключа /U можно командой UNFORMAT. Подробности работы с командой UNFORMAT приведены в ее описании.

Ключи

/V:*метка* указание метки тома (диска). Эта метка используется для идентификации конкретного диска и может включать в себя максимум 11 символов. В том случае, когда этот ключ опущен, DOS по окончании операции форматирования диска запросит у пользователя метку диска. Если вы форматируете несколько дисков одной командой FORMAT, все отформатированные диски будут иметь метку, указанную в ключе /V. Ключ /V несовместим в одной команде FORMAT с ключом /8. Для получения информации о метке тома следует просмотреть описания команд DIR, LABEL и VOL;

/Q быстрое форматирование диска. При указании этого ключа FORMAT обновляет корневой каталог и FAT на ранее отформатированном диске без проверки остального дискового пространства на дефекты. Используйте этот ключ для форматирования дисков, о которых известно, что они не имеют плохих участков;

/U проведение форматирования низкого уровня с физическим форматированием секторов и уничтожением информации на всем диске. Форматирование с ключом /U позволяет выявить сектора, в которых происходят ошибки чтения и записи информации, связанные с нарушением физической структуры носителя информации. Диск, отформатированный командой FORMAT с ключом /U, не может быть использован для восстановления информации командой UNFORMAT;

/F:*размер* указание размера форматируемого гибкого диска. Этот ключ можно использовать вместо ключей /T и /S. Допустимые значения параметра *размер*:

- 160 (или 160K, или 160KB), 160K, 1 сторона, одинарная плотность, 5.25-дюйм. диск;
- 180 (или 180K, или 180KB), 180K, 1 сторона, одинарная плотность, 5.25-дюйм. диск;
- 320 (или 320K, или 320KB), 320K, 2 стороны, двойная плотность, 5.25-дюйм. диск;
- 360 (или 360K, или 360KB), 360K, 2 стороны, двойная плотность, 5.25-дюйм. диск;
- 720 (или 720K, или 720KB), 720K, 2 стороны, двойная плотность, 3.5-дюйм. диск;
- 1200 (или 1200K, или 1200KB, или 1.2, или 1.2M, или 1.2MB), 1.2-MB, 2 стороны, высокая плотность, 5.25-дюйм. диск;
- 1440 (или 1440K, или 1440KB, или 1.44, или 1.44M, или 1.44MB), 1.44-MB, 2 стороны, высокая плотность, 3.5-дюйм. диск;
- 2880 (или 2880K, или 2880KB, или 2.88, или 2.88M, или 2.88MB) 2.88-MB, 2 стороны, сверхвысокая плотность, 3.5-дюйм. диск;

/B — резервирование дискового пространства для системных "скрытых" файлов MS-DOS - IO.SYS и MSDOS.SYS на новом отформатированном диске. В предыдущих версиях MS-DOS команда FORMAT в этом варианте применялось для того, чтобы на зарезервированное для системных файлов место потом перенести операционную систему командой SYS. Этот ключ поддерживается в MS-DOS версии 6.0 для обеспечения совместимости с предыдущими версиями DOS;

/S — при использовании этого ключа команда FORMAT после окончания форматирования диска переносит на него файлы операционной системы IO.SYS, MSDOS.SYS и COMMAND.COM. Этот прием используется для создания системного диска, то есть диска, с которого возможна загрузка операционной системы. Если команда FORMAT не сможет найти файлы ОС, тогда она попросит вставить системный диск;

/T:*дорожек* — указание числа дорожек на форматируемом диске. Когда имеется возможность, старайтесь использовать вместо этого ключа ключ /F. Совместное использование ключей /T и /F не допускается;

/N:*секторов* — указание количества секторов, расположенных на одной дорожке. Старайтесь по возможности заменять этот ключ ключом /F. Ключ */S:секторов* используется совместно с ключом */T:дорожек* для указания альтернативного метода задания типа форматируемой дискеты относительно ключа /F. Ключ /N нельзя совместно использовать с ключом /F;

/1 — форматирование только одной стороны гибких дисков;

/4 — форматирование дискеты емкостью 360 K на дисководе емкостью 1.2 MB. Когда этот ключ используется совместно с ключом /1, осуществляется форматирование дискеты емкостью 180 K на дисководе 1.2 MB;

/8 — форматирование 5.25-дюйм. дисков с 8-ю секторами на дорожке. Это ключ используется для обеспечения совместимости с дисками, отформатированными командой FORMAT версии MS-DOS 2.0 и ранее.

Примеры использования:

Для форматирования гибкого диска с максимальной плотностью, доступной для данного типа дисковода, введите команду:

```
format a:
```

При проведении быстрого форматирования диска A: используется следующий вариант команды FORMAT:

```
format a: /Q
```

Если вам необходимо отформатировать диск A: с физическим уничтожением всех данных, введите следующую команду:

```
format A: /U
```

В том случае, если вам необходимо отформатировать диск на емкость 360 K и потом сделать его системным, переписав файлы ОС, введите команду:

```
format a: /F:360 /S
```

Для форматирования диска A: с последующим присвоением ему метки "MS_DOS", введите следующую команду:

```
format a: /V:MS_DOS
```

Примечания:

- Если при вызове команды FORMAT вы не используете ключ /V, то после окончания форматирования, FORMAT выведет на дисплей следующее сообщение:

  ```
  Volume label (11 characters, ENTER for none)?
  Метка тома (11 символов, Enter не вводить)?
  ```

  Вы можете указать метку тома, используя максимум 11 символов, включая пробелы, либо не вводить ее, нажав клавишу Enter.

- При форматировании жестких дисков MS-DOS выводит на дисплей следующее сообщение:

  ```
  Warning, all data on non-removable disk
  drive x: will be lost!
  Proceed with Format (Y/N)?_
  (Предупреждение, все данные на несъемном диске
  диск x: будут уничтожены !
  Начать форматирование (Д/Н)?)
  ```

  Нажав клавишу Y, вы начнете форматирование жесткого диска; если процесс форматирования необходимо прервать, нажмите клавишу N.

- По окончании форматирования MS-DOS выводит на дисплей сообщения об общем объеме дискового пространства, объеме, занятом дефектными секторами, и объеме дискового пространства, занятого

файлами ОС (если в команде FORMAT использовались ключи /S и /B), а также количестве свободного дискового пространства.

- Рекомендуется при форматировании гибкого диска использовать ключ /U для того чтобы обнаружить дефектные области диска.
- Команда FORMAT не работает с дисками, созданными командой SUBST, сетевыми и электронными дисками, а также дисками, работающими под управлением INTERLNK.
- По окончании своей работы FORMAT возвращает DOS следующие коды завершения:

0	операция форматирования прошла успешно;
3	пользователь использовал клавиши Ctrl-C или Ctrl-Break для прекращения процесса форматирования диска;
4	в процессе форматирования произошла фатальная ошибка, отличная от ошибок с номерами 0, 3, 5;
5	пользователь использовал клавишу N для прерывания процесса форматирования диска, в ответ на запрос ОС: "Proceed with Format (Y/N)?" (Форматирование начать (Д/Н)?)

- Для дискет емкостью 160 K, 180 K, 320 K, 360 K допускается использование ключей /1, /4, /8, /B, /V, /S, /F. Для дискет емкостью 720 K, 1.2 MB, 1.44 MB, 2.88 MB допускаются ключи /B, /N, /T, /V, /S, /F, а для жестких дисков допускаются ключи /B, /V, /S.

GRAPHICS

Распечатывание графической копии экрана на принтере. Для использования команды GRAPHICS компьютер должен быть оснащен цветным или графическим адаптером. Команда внешняя. Файл GRAPHICS.COM. Команда GRAPHICS поддерживает все графические режимы видеоадаптеров CGA, EGA и VGA.

Формат команды:

graphics *тип диск:\маршрут\имя_файла* /R /B /LCD /PRINTBOX:STD|/PRINTBOX:LCD

Параметры:

тип - указание типа, подключенного к компьютеру принтера. Далее мы приводим те значения, которые может принимать этот параметр:

COLOR1 - IBM Personal Computer Color Printer с черной красящей лентой;

COLOR4 - IBM Personal Computer Color Printer с четырехцветной красящей лентой RGB (цвета - красный (R), зеленый (G), синий (B), черный). Эта лента позволяет получать на бумаге четырехцветное изображение;

COLOR8 - IBM Personal Computer Color Printer с четырехцветной красящей лентой CMY (цвета - бирюзовый (C), фиолетовый (M), желтый (Y), черный). Эта лента позволяет получать на бумаге восьмицветное изображение;

HPDEFAULT - любой принтер Hewlett-Packard, поддерживающей язык управления принтером PCL;

DESKJET - принтер Hewlett-Packard DeskJet;

GRAPHICS - группа графических принтеров IBM, включающая принтеры: IBM Personal Graphics Printer, IBM Proprinter, или IBM Quietwriter Printer и другие принтеры этих же серий. Тип этих принтеров установлен в команде GRAPHICS по умолчанию;

GRAPHICSWIDE - IBM Personal Graphics Printer с широкой кареткой 11 дюймов;

LASERJET - принтер Hewlett-Packard LaserJet;

LASERJETII - принтер Hewlett-Packard LaserJet II;

PAINTJET - принтер Hewlett-Packard PaintJet;

QUIETJET - принтер Hewlett-Packard QuietJet;

QUIETJETPLUS - принтер Hewlett-Packard QuietJet Plus;

RUGGEDWRITER - принтер Hewlett-Packard RuggedWriter;

RUGGEDWRITERWIDE - принтер Hewlett-Packard RuggedWriterWide;

THERMAL - принтер Thermal Printer для IBM PC-convertable;

THINKJET - принтер Hewlett-Packard ThinkJet.

диск:\маршрут\имя_файла - указание полного идентификатора файла, содержащего информацию о всех поддерживаемых командой GRAPHICS принтерах. Если этот параметр опущен, то MS-DOS считает, что файл GRAPHICS.PRO находится в текущем каталоге и в этом же каталоге находится файл GRAPHICS.COM.

Ключи

/R	печать изображения на принтере соответствующего изображению на экране компьютера (белые символы на черном фоне), в противном случае печать инверсного изображения (черные символы на белом фоне). По умолчанию этот ключ в команде GRAPHICS не используется;
/B	печать цветного фона. Этот ключ используется параметром, задающим тип принтера, который равен COLOR4 или COLOR8;

/LCD	вывод изображения, полученного на жидкокристаллическом дисплее (liquid crystal display - LCD) с коэффициентом формы, соответствующей коэффициенту формы дисплея CGA. Использование этого ключа имеет эффект, аналогичный использованию ключа /PRINTBOX:LCD;
/PRINTBOX:STD or /PRINTBOX:LCD	выбор размера графической копии экрана. Для правильного задания этого ключа следует посмотреть первый операнд в группе PRINTBOX вашего файла GRAPHICS.PRO. Если первый операнд этой группы равен STD, задавайте ключ /PRINTBOX:STD, а если первый операнд равен LCD, используйте ключ /PRINTBOX:LCD. Аббревиатуру PRINTBOX можно заменить ключом /PB.

Пример использования:

Для организации печати графического экрана на принтере введите следующую команду:

```
graphics
```

В момент, когда вам необходимо осуществить печать копии экрана на принтере, нажмите клавиши Shift-PrintScreen. MS-DOS просканирует информацию, выведенную на дисплей, и напечатает ее на принтере.

Примечания:

- Если печать экрана осуществляется в режиме среднего разрешения 320×200 точек и тип принтера установлен, как COLOR1 или GRAPHICS, команда GRAPHICS выводит на принтер изображение, имеющее 4 градации серого, соответствующее 4 цветам, видимым на экране. При распечатке графического экрана в режиме 640×200 команда GRAPHICS выводит изображение с поворотом на 90° градусов - альбомная ориентация (landscape orientation).
- Комбинация клавиш Shift-PrintScreen не может быть использована для распечатки изображений на принтерах, поддерживающих язык управления PostScript.
- Команда GRAPHICS загружается в память только один раз. Для выгрузки ее из памяти или запуска ее с другими параметрами и ключами необходимо перезапустить компьютер, а потом ввести команду GRAPHICS с требуемыми установками. Если же вы попытаетесь запустить команду GRAPHICS с новыми установками, не

перезагружая ваш компьютер, на дисплее появится следующее сообщение:

Unable to reload with profile supplied
(Запрещение перезагрузки с поддерживаемыми значениями)

HELP

Запуск справочной системы MS-DOS Help. Команда внешняя. Файл HELP.COM. После запуска программы экран компьютере приобретет вид, представленный на рисунке 8.6.

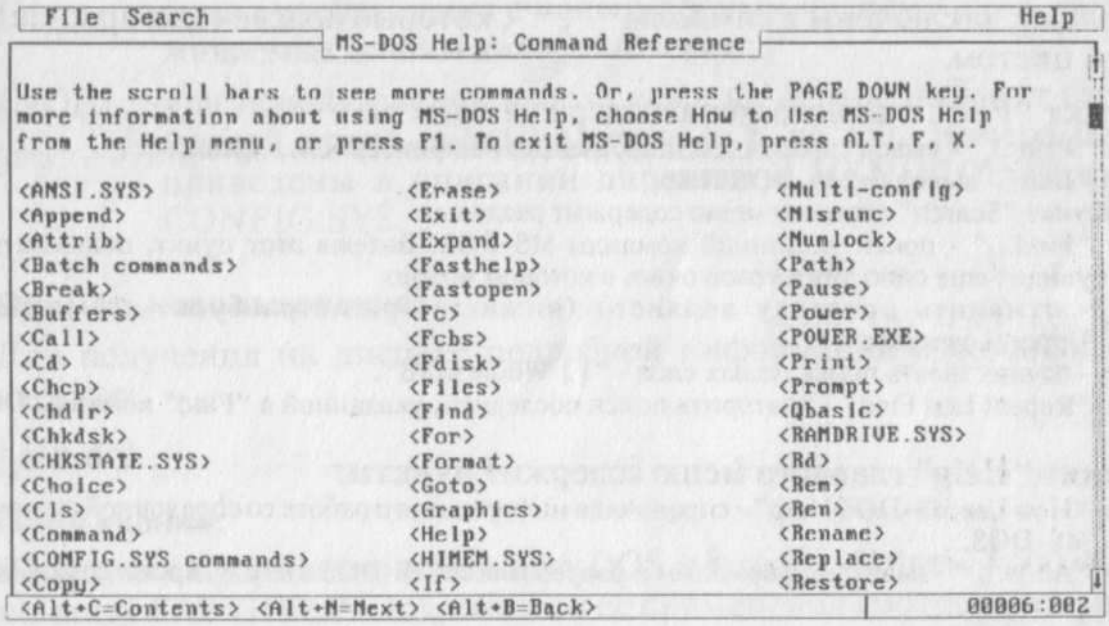


Рис. 8.6

Условно экран можно разделить на следующие части:

- строка меню, включающая в себя пункты: “File”, “Search”, “Help”;
- область экрана, содержащая подсказку о работе с системой помощи (перемещение по пунктам меню, прокрутка экрана, выход из утилиты). Область подсказки располагается сразу за строкой меню;
- область экрана, содержащая справочную информацию о командах ОС, директивах файла CONFIG.SYS, командах пакетных файлов и загружаемых драйверах устройств.

Управление справочной системой MS-DOS осуществляется при помощи клавиатуры, либо манипулятором типа “мышь”. Вход в меню программы осуществляется клавишей Alt.

Перемещение по пунктам системного меню возможно либо с помощью клавиш управления курсором, либо нажатием клавиш, соответствующих подсвеченным пунктам меню. Вход в меню программы и переме-

щение по пунктам программного меню можно также осуществить и с помощью "мыши", подведя ее указатель на необходимый пункт и нажав левую кнопку.

Для получения информации о команде DOS следует переместить курсор или указатель "мыши" на выбранную команду и нажать клавишу Enter, либо кнопку "мыши".

Далее MS-DOS выведет на экран информацию о назначении команды, ее формате (Syntax), примерах ее использования (Example) и примечаниях (Notes), относящихся к данной команде.

Все команды, директивы MS-DOS и т.д., по которым в Help имеется справка, заключены в символы "", "", которые подсвечены ярко-зеленым цветом.

Пункт "File" главного меню содержит пункты:

- "Print..." - вывод справочной информации на принтер или в файл;
- "Exit" - выход из MS-DOS Help;

Пункт "Search" главного меню содержит разделы:

- "Find..." - поиск указанной команды MS-DOS. Выбрав этот пункт, пользователь увидит еще одно диалоговое окно, в котором можно:
  - отменить проверку верхнего (нижнего) регистра букв - "[ ] Match Upper/Lowercase";
  - осуществлять поиск целых слов - "[ ] Whole word".
- "Repeat Last Find" - повторить поиск последней, указанной в "Find" команды DOS.

Пункт "Help" главного меню содержит пункты:

- "How Use MS-DOS Help" - справочная информация о работе со справочной системой MS-DOS;
- "About..." - вывод информации о разработчике MS-DOS Help и версии справочной системы.

Выбрав пункт "How Use MS-DOS Help", пользователь в появившемся новом окне может выбрать справку по следующим темам:

- "Navigation Thorough MS-DOS Help" - описание общих способов управления средой MS-DOS Help;
- "Searching for Text in MS-DOS Help" - поиск текста в среде MS-DOS Help;
- "Getting Context-Sensitive Help" - получение контекстно - чувствительной справки;
- "Using Dialog Box" - использование диалоговых окон;
- "Starting MS-DOS Help from Command line" - запуск MS-DOS Help из командной строки. Этот раздел подробно обсуждается далее;
- "Using MS-DOS Help using Keyboard" - управление средой MS-DOS Help с помощью клавиатуры.

Управление средой MS-DOS Help осуществляется также с помощью командной строки DOS.

Формат команды:

help /B /G /H /NOHI *тема*

Параметр:

тема - указание команды MS-DOS, по которой необходимо получение справочной информации.

Ключи

/B	вывод монохромного изображения на цветном мониторе. Используйте этот ключ также, если ваш компьютер оснащен цветным графическим адаптером и монохромным монитором;
/G	устранение эффекта "снега" для CGA - монитора;
/H	вывод изображения с максимальным числом линий, поддерживаемых вашим видеоадаптером;
/NOHI	при использовании этого ключа на экране не будут отображаться яркие цвета палитры (с 8 по 15). Номера цветов приведены в описании директивы MENUCOLOR файла CONFIG.SYS.

Пример использования:

Для получения на дисплее подробной информации о команде DEL, введите:

```
help del
```

Примечание:

Краткую информацию о команде DOS можно получить, использовав команду FASTHELP. Подробности ее применения смотрите в ее описании.

(!) INTERLNK

Переназначение разделяемых дисков двух компьютеров, соединенных с помощью последовательных или параллельных портов и работающих в локальной вычислительной сети (ЛВС) под управлением программ INTERLNK и INTERSVR.

В качестве примера можно привести следующий. Допустим, компьютер типа Notebook соединен кабелем с настольным компьютером и организована работа с разделяемыми файлами. Под разделяемыми файлами понимаются файлы, с которыми могут работать оба компьютера.

Для работы с этой командой необходимо в файле CONFIG.SYS подключить к ОС драйвер INTERLNK.EXE.

Команда внешняя. Файл INTERLNK.EXE.

Формат команды:

interlnk *клиент:* = *сервер:*

Для вывода на дисплей текущего статуса программы INTERLNK введите:

interlnk

Параметры:

клиент - указание буквы, обозначающей диск *клиента*, который должен быть переназначен в диск на *сервере* INTERLNK. Это диск должен быть одним из дисков, указанных при запуске программы INTERLNK.

сервер - указание буквы, обозначающей диск на *сервере* INTERLNK. Этот диск должен быть одним из дисков, указанных на экране с названием "This Computer (Server)", программы INTERLNK. Если в этом параметре не указана буква, то диск *клиента* не будет переназначен.

Пример использования:

Допустим, что на сервере запущена программа INTERSVR и на сервере имеется диск E. Для переназначения диска G: *клиента* в диск E: *сервера* необходимо ввести следующую команду:

interlnk g: = e:

Для отмены переназначения диска *клиента* F: введите следующую команду:

interlnk f: =

Примечания:

- Если вы используете программу INTERLNK для соединения с компьютером, на котором не загружена программа INTERSVR, утилита INTERLNK будет показывать подсоединение к клиенту 3-х добавочных накопителей на гибких дисках. Однако, если вы попытаетесь просмотреть содержимое дисков на этих накопителях, они окажутся пустыми. Для просмотра содержимого переназначенных дисков вы должны запустить на сервере утилиту INTERSVR.

  Когда вы соединяете два компьютера с использованием утилиты INTERLNK, каждый добавляемый диск получает в своем обозначении букву, следующую по алфавиту за буквой последнего доступного ОС логического диска. Используйте директиву LASTDRIVE файла CONFIG.SYS для установки числа дисков, которые может иметь ваш компьютер. Старайтесь выбрать число дисков в директиве LASTDRIVE таким, чтобы оно позволяло иметь в системе переназначенные диски.

- Для соединения двух машин при помощи последовательного порта используйте кабель с 9 или 25 контактными разъемами. В этом варианте работы три линии будут использованы для передачи данных: Земля-Земля (Ground-Ground), Передача-Прием (Transmit-Receive), Прием-Передача (Receive-Transmit). Всего для организации процесса передачи данных по последовательному интерфейсу используется семь линий. Разводка контактов для последовательного интерфейса (RS-232) приведена ниже.

1 компьютер			2 компьютер	
9 контактов	25 контактов		25 контактов	9 контактов
Контакт N:	Контакт N:		Контакт N:	Контакт N:
5 (Ground-Ground)		<——>	7	5
3 (Transmit-Receive)		<——>	3	2
7 (RTS - CTS)		<——>	5	8
6 (DSR - DTR)		<——>	20	4
2 (Receive-Transmit)		<——>	2	3
8 (CTS - RTS)		<——>	4	7
4 (DTR - DSR)		<——>	6	6

- При соединении двух машин с использованием параллельного интерфейса должна применяться следующая схема соединения с использованием разъемов DB-25:

1 компьютер		2 компьютер
25 контактов		25 контактов
Контакт N:		Контакт N:
2	<——>	15
3	<——>	13
4	<——>	12
5	<——>	10
6	<——>	11
15	<——>	2
13	<——>	3
12	<——>	4
10	<——>	5
11	<——>	6
25	<——>	25 (Ground-Ground)

(!) INTERSVR

Запуск "сервера" сети INTERLNK. Команда осуществляет загрузку в память компьютера программы, позволяющей организовать на вашем компьютере "сервер" для работы в простейшей ЛВС INTERLNK.

Команда внешняя. Файл INTERLNK.EXE.

Формат команды:

intersvr *диск:* /X=*диск:* /LPT:*n*|*адрес* /COM:*n:адрес* /BAUD:*скорость* /B /V

Для копирования файлов INTERLNK с одного компьютера на другой используйте следующий формат команды INTERLNK:

intersvr /RCOPY

Параметр:

диск: - указание буквы (букв), обозначающей переназначаемый диск (диски). По умолчанию переназначаются все диски.

Ключи

/X = *диск:*	указание буквы (букв), обозначающей диск (диски), не подлежащие переназначению;
/LPT:*n*|*адрес*	указание номера или адреса используемого параллельного порта. Если в ключе LPT используется параметр *n*, то необхо ,имо указывать номер параллельного порта. В случае, если параметр *n* не указан, "сервер" INTERLNK использует первый параллельный порт. Параметр *адрес* в ключе /LPT: используется при указании используемого порта через его адрес. Если в команде INTERSVR вы не укажете ни ключа /LPT, ни /COM, INTERLNK проверит наличие установленных на компьютере параллельных и последовательных портов и сам установит номера используемых портов;
/COM:*n*|*адрес*	указание номера или адреса используемого последовательного порта. Если в ключе COM используется параметр *n*, то необходимо указывать номер последовательного порта. В случае, если параметр *n* не указан, "сервер" INTERLNK использует первый последовательный порт. Параметр *адрес* в ключе /COM: используется при указании используемого порта через его адрес;

/BAUD:*скорость* установка скорости передачи данных по последовательному интерфейсу в бодах. Допустимыми значениями для этого параметра являются следующие: 9600, 19200, 38400, 57600 и 115200. По умолчанию скорость передачи данных установлена равной 115200 бод.

/B экран, отображающий состояние "сервера", будет черно-белым. Используйте этот ключ в том случае, если у вас имеются проблемы с выводом информации на черно-белый монитор;

/V предотвращение конфликтов при работе с системным таймером. Этот ключ необходимо использовать в том случае, если при соединении компьютеров по последовательному каналу один из них может быть остановлен во время работы программы INTERLNK с дисками или портом принтера;

/RCOPY копирование файлов утилиты INTERLNK с одного компьютера на другой. Компьютеры должны быть соединены нуль-модемным 7-жильным кабелем, и на компьютере, имеющем INTERLNK, должна иметься утилита MS-DOS MODE.

Примеры использования:

Допустим, что на каждом из двух соединенных компьютеров имеются логические диски A:, B:, C:. Для того, чтобы после запуска на одном из них утилиты INTERSVR его диски C:, A:, B: были соответственно переназначены в диски D:, E:, F:, на компьютере "клиента" достаточно ввести команду:

```
intersvr c: a: b:
```

Для того, чтобы при соединении компьютеров в вышеописанном примере диск A: не был переназначен и для соединения компьютеров использовался второй параллельный порт, следует ввести такую команду:

```
intersvr /x=a: /lpt2
```

Примечания:

- При переназначении дисков программа INTERLNK руководствуется следующим правилом: первый диск сервера будет переназначен на первый доступный диск клиента, второй диск сервера будет обозначен вторым доступным диском на машине "клиента" и т.д..

- Программы поддержки сети INTERLNK не переназначают сетевые диски, CD-ROM (лазерные компакт-диски) или другие диски, которые уже используют другие интерфейсы переназначения.
- При работе с последовательной "мышью" в среде Microsoft Windows и запуске в Windows "сервера" INTERLNK необходимо использовать для соединения компьютеров либо параллельный порт (ключ /LPT:), либо последовательный порт (ключ /COM:) с номером (адресом), отличным от номера (адреса) последовательного порта, используемого "мышью";
- При запуске "сервера" INTERLNK в многозадачной среде или в среде, допускающей переключение задач, комбинации клавиш или клавиша для переключения задачи могут быть запрещены. Для восстановления функций управления многозадачной средой необходимо выйти из "сервера" INTERLNK.
- При работе в сети INTERLNK запрещается загрузка с сервера и выполнение следующих команд DOS:

chkdsk	format
defrag	mirror
diskcomp	sys
diskcopy	undelete
fdisk	unformat

Для получения дополнительной информации об организации и работе с ЛВС INTERLNK необходимо просмотреть описание команды INTERLNK и драйвера INTERLNK.EXE в главе "Загружаемые драйверы устройств".

KEYB

Поддержка национальных кодовых страниц и клавиш, отличных от принятых в США, для клавиатуры компьютера. Команда внешняя. Файл KEYB.COM.

Формат команды:

При задании команды с использованием командной строки DOS используйте следующий формат:

keyb *xx,yyy,диск:\маршрут\имя_файла* /E /ID:*nnn*

Если же вы хотите выполнить команду KEYB на этапе обработки системой файла CONFIG.SYS, то команду KEYB следует задавать в следующем формате:

install=*диск1:\маршрут1*\keyb.com *xx,yyy,диск:\маршрут\имя_файла* /E /ID:*nnn*

Параметры:

xx - указание кода национальной клавиатуры. Для просмотра национальных кодов клавиатуры следует посмотреть таблицу в конце этого раздела.

yyy - указание кодовой таблицы. Для просмотра допустимых значений необходимо просмотреть таблицу в конце раздела. Если этот параметр не указан, то команда KEYB использует для него текущую кодовую страницу.

диск:\маршрут\имя_файла - указание полного идентификатора файла, содержащего информацию о кодовых страницах и символах национальных клавиатур. По умолчанию этим файлом является файл KEYBOARD.SYS. Если к файлу KEYBOARD.SYS уже указаны пути доступа, то этот параметр не является обязательным.

диск1:\маршрут1 - указание полного пути доступа к файлу KEYB.COM.

Ключи

/E — указание факта использования расширенной клавиатуры. Используйте этот ключ для работы с расширенной клавиатурой на компьютерах, оснащенных процессором Intel 8086 или совместимых с ним;

/ID:*nnn* — указание типа клавиатуры. Этот ключ применяется в случае, если в одной стране применяется более одного типа клавиатуры, например, во Франции, Италии и Великобритании. Для правильного использования этого ключа необходимо просмотреть таблицу с данными по национальным клавиатурам и кодовым страницам.

Таблица 8.1

Национальные кодовые таблицы и клавиши

Страна или язык	Код клавиатуры (xx)	Кодовая страница (yy)	Тип клавиатуры (IDnnn)
Бельгия	BE	850, 437	
Бразилия	BR	850, 437	
Канада (франц.)	C	850, 863	
Чехия	CZ	852, 850	

Национальные кодовые таблицы и клавиши			
Страна или язык	Код клавиатуры (xx)	Кодовая страница (yy)	Тип клавиатуры (IDnnn)
Словакия	SL	852, 850	
Дания	DK	850, 865	
Финляндия	SU	850, 437	
Франция	FR	850, 437	120, 189
Германия	GR	850, 437	
Венгрия	HU	852, 850	
Италия	IT	850, 437	141, 142
Латин. Америка	LA	850, 437	
Нидерланды	NL	850, 437	
Норвегия	NO	850, 865	
Польша	PL	852, 850	
Португалия	PO	850, 860	
Испания	SP	850, 437	
Швеция	SV	850, 437	
Швейцария (франц.)	SF	850, 437	
Швейцария (немец.)	SG	850, 437	
Великобритания	UK	850, 437	166, 168
США	US	850, 437	
Югославия	YU	852, 850	

Пример использования:

Для использования немецкой клавиатуры на вашем компьютере при нахождении файла KEYBOARD.SYS в каталоге \DOS диска C: введите команду:

```
keyb gr,,c:\dos\keyboard.sys
```

Примечания:

- Кодовая страница, которую вы указываете в параметре *yyy*, должна быть доступна на вашем компьютере. В противном случае все символы, не входящие в стандартную кодовую страницу, не будут использоваться.

- Команда KEYB при вводе ее без параметров и ключей выводит на дисплей текущие установки для клавиатуры и кодовой страницы. Сообщение команды KEYB в этом случае имеет вид:

  Current keyboard code: FR code page: 437
  Current CON code page: 437

- Для переключения с национальной кодовой страницы на стандартную кодовую страницу нужно воспользоваться комбинацией клавиш Ctrl-Alt-F1. Для возвращения к установленным параметрам клавиатуры и кодовой страницы используйте комбинацию клавиш Ctrl-Alt-F2. Для переключения режима "typewriter mode" на стандартный используйте клавиши Ctrl-Alt-F7.

- Команда KEYB по окончании своей работы возвращает в MS-DOS следующие коды завершения:

  0 файл с данными по национальной клавиатуре и кодовым страницам успешно загружен;
  1 неправильный код клавиатуры, кодовой страницы или неправильный формат задания команды;
  2 файл с данными по национальным клавиатурам и кодовым таблицам является плохим или не содержит сведений для поддержки национальных клавиатур;
  4 ошибка обнаружена во время ввода данных с клавиатуры или при их выводе на монитор;
  5 используемый символ принадлежит неподготовленной кодовой странице.

  Коды завершения команды KEYB могут использоваться в командных файлах (команда IF ERRORLEVEL) или прикладными программами.

LABEL

Создание, изменение или удаление метки диска (тома). При выводе листинга каталогов MS-DOS выводит в начале листинга метку диска. Метка диска выводится также и другими командами MS-DOS (VOL, FDISK и др.) Если диск имеет кроме метки тома и серийный номер, он также включается в выходную информацию указанных команд DOS. Команда внешняя. Файл LABEL.EXE.

Формат команды:

label *диск: метка*

В случае, если вам необходимо посмотреть текущую метку диска и его серийный номер, введите команду LABEL в виде:

label

После того, как MS-DOS выведет на дисплей метку диска и его серийный номер (если он существует), пользователь может в ответ на запрос MS-DOS модифицировать метку или удалить ее. Для изменения метки достаточно ввести новую метку и затем нажать клавишу ENTER, а для удаления текущей метки - нажать клавишу ENTER, и на вопрос MS-DOS "Delete current volume label (Y/N) ?" ответить Y.

Параметры:

диск: - указание буквы, обозначающей диск, с которым будет работать команда LABEL.

метка - указание новой метки диска. Метка и буква диска обязательно должны разделяться символом ":".

Пример использования:

Для изменения текущей метки на диске C: на метку "MS_DOS6_0", необходимо ввести следующую команду:

```
label c:ms_dos6_0
```

Примечания:

- При вводе команды LABEL на экране появится сообщение следующего формата:

  ```
  Volume in drive A is XXXXXXXXXXX
  Volume Serial Number is XXXX-XXXX
  Volume label (11 characters, ENTER for none)?
  (Том диска имеет имя XXXXXXXXXXX)
  (Серийный номер тома XXXX-XXXX)
  (Метка тома (11 символов, ENTER для отмены)?)
  ```

- При указании метки диска можно использовать 11 символов, включая пробелы, но без символов табуляции. Несколько пробелов будут преобразованы в одиночный пробел. При указании метки тома нельзя использовать следующие символы:

  "*", "?", "/", "\", "|", ".", ",", ";", ":", "+", "=", "[", "]", "(", ")", "&", "^", "<", ">"

  MS-DOS выводит метку диска заглавными буквами. Если вы вводите метку диска прописными буквами, утилита LABEL автоматически преобразует метку диска, и выведет ее заглавными буквами.

- Утилита LABEL не работает с дисками, созданными командой SUBST.

(!) LOADFIX

Обеспечение гарантированной загрузки программы в первые 64K основной памяти и запуск ее на выполнение.

Команда внешняя. Файл LOADFIX.EXE.

Формат команды:

loadfix *диск:\маршрут\имя-файла параметры-программы*

Параметры:

диск:\маршрут - указание местоположения загружаемого файла.

имя-файла - указывает имя загружаемого под управлением LOADFIX выполняемого модуля.

параметры-программы - указание параметров командной строки, передаваемых загружаемой программе.

Пример использования:

Для загрузки программы PRIMER.EXE гарантированно в первые 64 K основной памяти (программа находится в каталоге \PAS диска C:) введите команду:

```
loadfix c:\pas\primer.exe
```

Команду LOADFIX удобно использовать в файле AUTOEXEC.BAT для загрузки различных сервисных программ.

Примечания:

- Некоторые программы могут выводить сообщение типа:

  Packed file corrupt
  (Упакованные файлы испорчены)

- Это сообщение об ошибке возникает в том случае, если вся программа или ее часть загружена в первые 64 K основной памяти, и она не может быть успешно выполнена. Ошибка наиболее вероятно связана с тем, что при загрузке драйверов устройств в область старшей памяти программа не может нормально работать с ними.
- При выводе MS-DOS вышеописанных сообщений для гарантированной загрузки таких программ в область первых 64 K основной памяти используйте утилиту LOADFIX.

LOADHIGH (LH)

Загрузка программ в область старшей памяти. Эта команда дает возможность увеличить объем основной памяти, доступной другим программам. Команда внутренняя.

Формат команды:

loadhigh *диск:\маршрут\имя_файла параметры*

Для указания области памяти, в которую вы хотите загрузить программу, используйте следующий формат команды:

loadhigh /L:*область1,мин_размер1;область2,мин_размер2,...* /S *диск:\маршрут\имя_файла параметры*

Вместо аббревиатуры LOADHIGH возможно использование символов LH.

Параметры:

диск:\маршрут\имя_файла - указание полного идентификатора файла, содержащего загружаемую в память программу.

параметры - указание параметров командной строки, передаваемых загружаемой в старшую память программе.

Ключи

/L:*область1, мин_размер1; область2, мин_размер2,...*	указание одной или нескольких областей старшей (UMB) памяти, которые будут использованы для загрузки программы. В том случае, если ключ /L в команде LOADHIGH не используется, MS-DOS загружает программу в свободные блоки старшей памяти и делает все блоки UMB доступными загруженной программе. Ключ /L может быть использован для загрузки программ в специфические области памяти или для явного указания областей памяти, которые будут использованы программой. Для примера, если необходимо загрузить программу в свободный блок старшей памяти с номером 4, вы должны использовать ключ /L:4. Для просмотра свободных блоков старшей памяти используйте команду MEM с ключом /F. Несколько программ, загруженных командой LOADHIGH с ключом /L, не могут одновременно использовать одну и ту же область памяти. Поэтому необходимо для каждой программы указывать только отведенную ей область памяти. Для просмотра

того, как программы используют память, выполните команду MEM с ключом /M.

Обычно MS-DOS загружает программу в указанную область памяти, которая будет равна размеру выполняемого файла. В том случае, ели DOS предоставляет область старшей памяти размером, не позволяющим разместить полностью выполняемый модуль, необходимо в ключе /L использовать параметр *мин_размер* для указания размера старшей памяти, которая будет гарантированно выделена загружаемой программе;

/S сжатие размера UMB до минимума, пока производится загрузка программы. Используйте этот ключ для более эффективного использования оперативной памяти. Этот ключ обычно используется в команде LH, заданной с помощью утилиты DOS MemMaker. Программа MemMaker анализирует программное обеспечение, функционирующее на вашем компьютере и проводит его оптимизацию. Для оптимизации использования оперативной памяти часто используется команда LH, вводимая MemMaker в состав файла AUTOEXEC.BAT.

Примеры использования:

Для загрузки программы DOSKEY в область старшей памяти номер 1, которая находится на диске C: в каталоге \DOS, используете команду LOADHIGH в следующем варианте:

```
loadhigh /L:1 c:\dos\doskey
```

В том случае, если вы хотите, чтобы программа PRIMER.EXE использовала блок номер 1 старшей памяти, а также блоки 3 и 4 UMB, используйте команду:

```
lh /L:1;3;4 c:\app\program.exe
```

Если вам необходимо, чтобы программа PRIMER использовала основную память (область 0) и блок старшей памяти 1, необходимо применить команду LH в следующем варианте:

```
loadhigh /L:0;1 c:\app\program.exe
```

Примечания:

- Для того, чтобы использовать команду LOADHIGH, вы должны использовать директиву DOS=UMB в вашем файле CONFIG.SYS. Для получения дополнительной информации о директиве DOS фай-

ла CONFIG.SYS необходимо посмотреть ее описание в главе "Директивы файла CONFIG.SYS".

- Для того, чтобы вы смогли загрузить программу в область старшей памяти, необходимо использовать программу-менеджер старшей памяти. Для этих целей в MS-DOS имеется драйвер EMM386, который можно использовать на машинах с процессором 80386 или выше. Директива DEVICE загрузки этого драйвера должна стоять после директивы DEVICE, загружающей драйвер управления дополнительной памятью HIMEM.SYS.
- При загрузке программ командой LOADHIGH в область старшей памяти, MS-DOS предоставляет им всю доступную старшую память. В случае нехватки места для программы в области старшей памяти DOS загрузит программу в основную память. Для определения того, как программа использует старшую память, используйте команду MEM с ключом /M и указанием в качестве аргумента имени загруженной программы.

MEM

Просмотр информации об использовании программами оперативной памяти компьютера. Команда внешняя. Файл MEM.EXE. Команда MEM используется для вывода на дисплей общего объема оперативной памяти машины, объема памяти, свободной от прикладных программ, и размере памяти, занимаемой программами, загруженными в текущий момент.

Формат команды:

mem /CLASSIFY|/DEBUG|/FREE|/MODULE *имя_программы* /PAGE

Для вывода информации о том, как используется в вашем компьютере память, используйте команду MEM в следующем виде:

```
mem
```

Ключи

/CLASSIFY	вывод на дисплей списка программ, загруженных в память в текущий момент времени в сокращенном виде. На экран выводится список программ, загруженных как в основную, так и в старшую память. Команда MEM /CLASSIFY выводит на дисплей также общий объем оперативной памяти и объем памяти, которая свободна в данный момент времени.

Ключ /CLASSIFY может совместно использоваться с ключом /P. Вместо ключа /CLASSIFY допускается использование ключа /C;

/DEBUG — добавление в выходную информацию команды MEM информации о внутренних драйверах устройств DOS (CLOCK$, CON, AUX и др.). Команда MEM /DEBUG выводит на дисплей имя каждого модуля, сегментные адреса, тип модуля и общую информацию о распределении оперативной памяти.

Ключ /DEBUG допускается использовать с ключом /P и записывать его в виде /D;

/FREE — вывод на дисплей объема свободных областей основной и старшей памяти. Команда MEM /FREE выводит на дисплей сегментные адреса и размер свободной области основной памяти, а также показывает размер свободного блока старшей памяти в каждой области старшей памяти.

Вы можете использовать ключ /FREE совместно с ключом /PAGE. Вместо ключа /FREE допускается использование ключа /F;

/MODULE *имя_программы* — вывод информации о том, как указанный в параметре *имя_программы* модуль использует оперативную память компьютера. Имя модуля должно указываться после ключа /MODULE. MEM /MODULE показывает области памяти, которые использует программа, а также выводит информацию о сегментных адресах и размере каждой используемой области памяти.

Ключ /MODULE допускается заменять ключом /M и использовать вместе с ключом /PAGE;

/PAGE — организация паузы при выводе информации на экран дисплея. После того, как экран дисплея (24 строки) полностью заполнен, организуется пауза до нажатия любой клавиши.

Примеры использования:

Допустим, ваш компьютер имеет расширенную (Expanded) и дополнительную (Extended) память. Для вывода на дисплей информации об основной, расширенной, дополнительной памяти, а также о программах, загруженных в память, необходимо ввести команду:

```
mem /classify
```

Результаты отчета команды MEM будут иметь вид:

```
Modules using memory below 1 MB:
(Модули, использующие свыше 1 MB:)

Name          Total        =   Conventional    +   Upper Memory
(Имя)        (Общая)           (Основная)          (Старшая)
________  ________________   ________________   ________________
MSDOS       18381   (18K)      18381   (18K)          0    (0K)
SETVER        624    (1K)        624    (1K)          0    (0K)
HIMEM        1104    (1K)       1104    (1K)          0    (0K)
EMM386       4096    (4K)       4096    (4K)          0    (0K)
PROTMAN       128    (0K)        128    (0K)          0    (0K)
UBXPS       98624   (96K)      98624   (96K)          0    (0X)
SMARTDRV    29296   (29K)       2480    (2K)      26816   (26K)
DBLSPACE    47184   (46K)      47184   (46K)          0    (0K)
COMMAND      3168    (3K)       3168    (3K)          0    (0K)
WIN386      81312   (79K)      20880   (20K)      60432   (59K)
PROTMAN      2560    (3K)       2560    (3K)          0    (0K)
WIN          1584    (2K)       1584    (2K)          0    (0K)
COMMAND      3376    (3K)       3376    (3K)          0    (0K)
MOUSE       17328   (17K)          0    (0K)      17328   (17K)
UBNEI        6704    (7K)          0    (0K)       6704    (7K)
WORKGRP      4400    (4K)          0    (0K)       4400    (4K)
SHARE        6208    (6K)          0    (0K)       6208    (6K)
DOSKEY       4144    (4K)          0    (0K)       4144    (4K)
Free       451264  (441K)     451264  (441K)          0    (0K)

Memory Summary:
(Общая память:)

Type of Memory         Total     =      Used       +      Free
(Тип памяти)          (Общая)       (Используется)     (Свободная)
________________  _______________  ________________  ________________
Conventional       655360  (640K)   204096  (199K)   451264  (441K)
Upper              126032  (123K)   126032  (123K)        0    (0K)
Adapter RAM/ROM    267184  (261K)   267184  (261K)        0    (0K)
Extended (XMS)    4194304 (4096K)  3145728 (3072K)  1048576 (1024K)
Expanded (EMS)          0    (0K)        0    (0K)        0    (0K)
________________  _______________  ________________  ________________
Total memory      5242880 (5120K)  3743040 (3655K)  1499840 (1465K)

Total under 1 MB   781392  (763K)   330128  (322K)   451264  (441K)

Largest executable program size          451248    (441K)
(Размер выполняемого модуля)
Largest free upper memory block               0      (0K)
(Размер свободного блока старшей памяти)
MS-DOS is resident in the high memory area.
(MS-DOS резидента в старшей области памяти)
```

Пункт "Adapter RAM/ROM" содержит сведения о "теневой" BIOS. "Теневая" BIOS представляет собой область памяти, в которую скопированы данные ROM BIOS для ускорения работы компьютера.

Строка "Largest executable program size" содержит сведения о максимальном размере программы, которую можно выполнить.

В строке "Largest free upper memory block" приведен размер максимальной области старшей памяти, доступной прикладным программам.

"MS-DOS is resident in the high memory area" - эта строка сообщает пользователю, что MS-DOS использует для работы первые 64K старшей памяти.

Для получения информации об использовании памяти отдельной программой необходимо использовать следующий формат команды MEM:

```
mem /MODULE win386
```

Приведенный пример показывает, как модуль WIN386 использует оперативную память компьютера. Листинг команды MEM в таком варианте ее использования имеет следующий вид:

```
 Segment   Region       Size          Type
(Сегмент) (Область)   (Размер)       (Тип)
 ______   ______     ____________   _________
  005D2                  64    (0K)   Data
  00D02                 160    (0K)   Data
  00EF7                 240    (0K)   Environment
  00F07               28592   (27K)   Program
  0D7C0       1         976    (1K)   Data
  0EEBF       2        5120    (5K)   Data
                      ____________
 Total size: 35152   (33K)
```

Листинг содержит сведения о сегментных адресах и размере каждой области памяти, занимаемой программой, и о типе данных в каждой области памяти. В конце листинга дается отчет об общем размере исследуемой программы (33K).

Примечания:

- Команда DOSKEY позволяет добавлять в каждую команду MEM ключ /PAGE, для того чтобы сделать работу с выходной информацией команды MEM более удобной. Описанная выше возможность команды DOSKEY реализуется, например, добавлением в состав вашего файла AUTOEXEC.BAT следующих строк:

```
c:\dos\doskey
doskey mem = mem.exe $* /p
```

- MS-DOS выводит на дисплей статус расширенной памяти в том случае, если ваша система имеет свыше одного мегабайта памяти, и расширенная память отвечает спецификации LIM EMS 4.0. Необходимым условием получения сведений о расширенной памяти является также подключение к ОС драйвера EMS-памяти EMM386.

- Статус расширенной памяти не будет выведен, если вы попытаетесь выполнить команду MEM, находясь в среде Microsoft Windows версии 3.0.

(!) MEMMAKER

Автоматическая оптимизация использования оперативной памяти, занятой системными и прикладными программами, размещенными резидентно в основной и старшей памяти. Для использования программы MemMaker необходимо наличие компьютера с процессором 80386, 80486 или Pentium и расширенной (Extended) памяти. Команда внешняя. Файл MEMMAKER.EXE.

После запуска MemMaker экран компьютера примет вид, аналогичный приведенному на рисунке 8.7.

```
Microsoft MemMaker
-------------------------------------------------------------------------------

Welcome to MemMaker.

MemMaker optimizes your system's memory by moving memory-resident
programs and device drivers into the upper memory area. This
frees conventional memory for use by applications.

After you run MemMaker, your computer's memory will remain
optimized until you add or remove memory-resident programs or
device drivers. For an optimum memory configuration, run MemMaker
again after making any such changes.

MemMaker displays options as highlighted text. (For example, you
can change the "Continue" option below.) To cycle through the
available options, press SPACEBAR. When MemMaker displays the
option you want, press ENTER.

For help while you are running MemMaker, press F1.

              Continue or Exit? Continue

ENTER=Accept Selection  SPACEBAR=Change Selection  F1=Help  F3=Exit
```

Рис. 8.7

Интерфейс программы имеет достаточно удобный вид. Верхняя часть программы содержит описание тех действий, которые собирается предпринять программа MemMaker.

В нижней части экрана подсвечивается вариант ответа пользователя на вопрос MemMaker. Нижняя часть экрана содержит подсказки пользователю по работе с MemMaker. Работать с MemMaker не сложнее, чем с любой командой DOS, и даже начинающие пользователи MS-DOS сумеют овладеть этой командой. В ходе своей работы MemMaker перезагружает компьютер, что является совершенно нормальным действием.

Команда внешняя. Файл MEMMAKER.EXE. Команда не может быть использована при работе в среде Microsoft Windows.

Формат команды:

memmaker /B /BATCH /SESSION /SWAP:*диск* /T /UNDO /W:*n,m*

Ключи

/B	вывод информации о работе MemMaker в черно-белом режиме. Используйте этот ключ, если MemMaker некорректно отображает информацию на дисплее;
/BATCH	запуск MemMaker в командном режиме. В командном режиме MemMaker сам автоматически отвечает на все вопросы. В случае обнаружения ошибок MemMaker восстанавливает старые файлы CONFIG.SYS и AUTOEXEC.BAT, а также, если это необходимо, файл Windows SYSTEM.INI. После окончания работы, MemMaker создает файл MEMMAKER.STS, в котором пользователь имеет возможность просмотреть статус сообщений MemMaker. Для просмотра файла MEMMAKER.STS можно использовать любой текстовый редактор, например MS-DOS Editor или команду TYPE;
/SESSION	работа только утилиты MemMaker для организации процесса оптимизации;
/SWAP:*диск*	указание буквы диска, обозначающей ваш оригинальный системный диск. Этот ключ используется в том случае, если известно, что ваш системный диск будет изменен при запуске компьютера. Такая ситуация может возникнуть при использовании программ "сжатия" дисков, так в случае изменения "горячего" диска может измениться информация и на "сжатых" дисках. Если буква, обозначающая ваш системный диск, или этот ключ, не используется, то MemMaker будет сам искать местоположение ваших системных файлов. При использовании программ Stacker 2.0, Superstor, DoubleSpace этот ключ использовать нельзя;
/T	запрещение определения ЛВС IBM Token-Ring. Необходимо применять этот ключ в том случае, если ваш компьютер входит в состав ЛВС, и вы не хотите иметь дополнительных проблем при работе MemMaker;
/UNDO	восстановление предшествующих работе MemMaker установок ОС. Когда MemMaker оптимизирует память

вашей системы, он вносит изменения в файлы CONFIG.SYS и AUTOEXEC.BAT, а при необходимости - и в состав файла Windows - SYSTEM.INI. Если внесенные MemMaker изменения не удовлетворяют вас, или после работы MemMaker ваша система не работает, то, используя команду MemMaker с ключом /UNDO, можно восстановить исходные установки операционной системы;

/W:*n*,*m* указание областей старшей памяти, резервируемой для буферов трансляции Windows. Windows использует для буферов трансляции две области старшей памяти. Параметр *n* указывает на первую область старшей памяти, а значение *m* на вторую область старшей памяти. По умолчанию Memmaker не отводит в памяти место для буферов трансляции Windows. В этом случае параметры *n* и *m* равны нулю.

Примеры использования:

Для запуска MemMaker в командном режиме введите команду:

```
memmaker /batch /w:0,0
```

Для восстановления утилитой MemMaker предыдущих установок вашей системы необходимо ввести команду:

```
memmaker /undo
```

Допустим, на вашем компьютере запущена программа "сжатия" дисков. Ваш системный диск имеет обозначение C:. После работы программы "сжатия" диск C: станет главным сжатым диском, а системные файлы будут размещаться на диске D:. Для работы с такими дисками необходимо использовать утилиту MemMaker в следующем формате:

```
memmaker /swap:d
```

Эта команда указывает на оригинальный системный диск D:, который хранит файлы операционной системы и файлы CONFIG.SYS и AUTOEXEC.BAT.

Примечания:

- Для получения дополнительной информации о загрузке драйверов в старшую память необходимо просмотреть описание директивы DEVICEHIGH в главе "Директивы файла CONFIG.SYS".
- О том, как загружать программы в область старшей памяти, подробно рассказывает описание команды LOADHIGH (LH).

MKDIR (MD)

Создание каталога. Команда внутренняя. Команда MKDIR или MD может использоваться для создания многоуровневой древовидной структуры каталогов.

Формат команды:

mkdir *диск:\маршрут*
md *диск:\маршрут*

Параметры:

диск: - указание логического диска, на котором предполагается создание каталога.

маршрут - указание имени и местонахождения нового создаваемого каталога.

Максимальная длина всех одиночных путей из корневого каталога в новый каталог может составлять 63 символа, включая символы "\".

Пример использования:

Для создания каталога на текущем диске, в текущем каталоге, необходимо использовать команду MKDIR без указания параметров *диск:\маршрут*. Для создания каталога с именем BOOKS введите команду:

```
mkdir books
```

Этих же результатов можно добиться использованием команды:

```
md books
```

В том случае, когда вам необходимо в созданном ранее каталоге, например BOOKS, создать новый каталог, необходимо предпринять следующие шаги:

- сделать каталог BOOKS текущим, например командой CD;

- ввести команду:

```
mkdir dictionary
```

После выполнения этой команды будет создан каталог DICTIONARY, подчиненный каталогу BOOKS.

MODE

Задание конфигурации для системных драйверов. Команда MODE решает несколько задач, таких как установка параметров принтера, последовательных портов, режимов работы клавиатуры и дисплея. Так

как команда MODE решает несколько задач, для работы с каждым из поддерживаемых драйверов необходим свой формат. Команда внешняя. Файл MODE.COM.

Форматы команды MODE для различных режимов работы приведены далее.

УСТАНОВКА ХАРАКТЕРИСТИК ПРИНТЕРА, ПОДСОЕДИНЕННОГО С ПОМОЩЬЮ ПАРАЛЛЕЛЬНОГО ПОРТА

В этом варианте работы команда MODE позволяет установить характеристики IBM-совместимых и Epson-совместимых принтеров, подключенных через параллельный порт (LPT1, LPT2, LPT3).

Форматы команды:

mode LPT*n*: *c*,*l*,*r*
mode LPT*n*: COLS=*c* LINES=*l* RETRY=*r*

Параметры:

LPT*n* - указание номера параллельного порта, к которому подсоединен принтер. Допустимыми значениями для параметра *n* лежат в пределах от 1 до 3. Когда в команде MODE LPT опущены один из параметров c, l, r, команда MODE сама установит опущенный параметр по умолчанию. При использовании короткой формы задания команды (без использования слов COLS=, LINES=, RETRY=), утилита MODE сама распознает параметры в соответствующих позициях. Однако следует помнить, что если при короткой форме записи команды MODE вы опускаете один из параметров, необходимо вместо него использовать символ "," перед следующим параметром;

COLS=*c* - указание числа символов в строке: 80 или 132. По умолчанию принимается 80 символов в строке. Параметр COLS= можно опустить и использовать только значение *c*;

LINES=*l* - указание числа строк межстрочного интервала: 6 или 8. Измеряется в числе строк на дюйм. Вместо параметра LINES= возможно использование значения вместо *l*;

RETRY=*r* - указание типа реакции системы при обнаружении ОС состояния тайм-аута. Этот параметр определяет также способ размещения резидентной части MODE в оперативной памяти. Параметр *r* может принимать следующие значения:

B - возвращение состояния "занято" при проверке статуса занятого порта;

E - возвращение ошибки при проверке статуса занятого порта;

P - повторение передачи данных на принтер, когда он выключен;

R - возвращение состояния "готово" при проверке статуса занятого порта;

N - прекращение печати при обнаружении занятого принтера. Это значение установлено по умолчанию. Вместо N допускается использовать аббревиатуру NONE.

Параметр *r* нельзя использовать при работе с сетевыми принтерами.

Примеры использования:

Для установки на принтере, подключенном к LPT2, длины строки в 80 символов и межстрочного интервала в 1/8 дюйма необходимо ввести команду:

```
mode lpt2:80,8
```

Для организации режима печати, при котором ОС будет повторять попытки печати файла до тех пор, пока файл не будет напечатан, введите команду MODE LPT:n в следующем формате:

```
mode lpt2:,8,b
```

Если вам необходимо остановить процесс печати, во время попыток ОС вывести на печать указанный файл, нажмите комбинацию клавиш Ctrl-Break или используйте команду MODE без параметра *r*.

Примечания:

- Использование параметра RETRY=r имеет тот же смысл, что и использование параметра P в предыдущих версиях MS-DOS.
- Использование команды MODE LPT:n отменяет действие команды MODE LPT:n = COM:m, о работе которой рассказано далее.

УСТАНОВКА РАБОЧИХ ПАРАМЕТРОВ ПОСЛЕДОВАТЕЛЬНЫХ ПОРТОВ

Этот вариант использования утилиты MODE поддерживает следующие последовательные порты: COM1, COM2, COM3, COM4.

Форматы команды:

mode COM*m*: *b*,*p*,*d*,*s*,*r*
mode COM*m*: BAUD=*b* PARITY=*p* [DATA=*d* STOP=*s* RETRY=*r*

Параметры:

COM*m* - задание номера последовательного порта, для которого будет производиться установка рабочих параметров. Допустимые значения для *m* располагаются в пределах от 1 до 4. Если в команде MODE COM*m*: вы не укажите один из пяти параметров, утилита MODE сама настроит эти параметры. При использовании короткой формы задания параметров (без использования слов BAUD=, PARITY=, DATA= и т.д.),

MODE сама распознает указанные в короткой форме значения. Следует только помнить, что если вы не используете параметры, то вместо них перед следующим параметром должна стоять запятая.

BAUD=*b* - задание скорости передачи данных в бодах, с помощью первых двух цифр, соответствующих задаваемой скорости передачи данных. Вместо параметра BAUD= допустимо использование значения вместо символа *b*. Параметр BAUD= может принимать следующие значения:

11	110 бод	24	2400 бод
15	150 бод	48	4800 бод
30	300 бод	96	9600 бод
60	600 бод	19	19200 бод
12	1200 бод		

Скорость передачи в 19200 бод поддерживается не всеми типами компьютеров, поэтому, прежде чем использовать команду MODE с этим параметром, проверьте оборудование вашего компьютера;

PARITY=*p* - задание типа контроля данных (паритета) при обмене информацией через последовательный порт. Допускаются следующие значения этого параметра:

N (none) - нет паритета;

E (even) - "четный" паритет;

O (odd) - "нечетный" паритет;

M (mark) - маркер;

S (space) - пробел.

По умолчанию установлен четный паритет - E. Не все компьютеры поддерживают значения для паритета M и S. Вы можете вместо аббревиатуры PARITY использовать значение, обозначенное символом *p*;

DATA=*d* - задание числа информационных битов в одном кадре асинхронной передачи. Число бит меняется в пределах от 5 до 8. По умолчанию это значение равно 7. Не все компьютеры поддерживают значения 5 и 6 для описываемого параметра. Вместо параметра DATA= возможно использование значения, помещенного вместо символа *d*;

STOP=*s* - указание числа стоповых битов в кадре асинхронной передачи. Параметр может принимать следующие значения: 1, 1.5, 2. По умолчания для скорости передачи 110 бод этот параметр равен 2. Для других скоростей передачи он равен 1. Не все компьютеры поддержи-

вают значение 1.5. Параметр STOP= можно опустить и использовать вместо него значение, введенное в команду вместо символа *s*.

RETRY=*r* - указание типа реакции системы при обнаружении ОС состояния тайм-аута последовательного порта. Этот параметр определяет также способ размещения резидентной части MODE в оперативной памяти. Параметр *r* может принимать следующие значения:

B - возвращения состояния "занято" при проверке статуса занятого порта;

E - возвращение ошибки при проверке статуса занятого порта;

P - повторение передачи данных на принтер, когда он выключен;

R - возвращение состояния "готово" при проверке статуса занятого порта;

N - прекращение печати при обнаружении занятого принтера. Это значение установлено по умолчанию. Вместо N допускается использовать аббревиатуру NONE.

Параметр *r* нельзя использовать при работе с сетевыми принтерами.

Пример использования:

```
mode com2: 48
```

Для второго последовательного порта COM2 устанавливается скорость передачи данных 4800 бод.

Примечания:

Параметр RETRY=B команды MODE имеет то же значение, что и параметр P в предыдущих версиях DOS.

ВЫВОД НА ДИСПЛЕЙ ТЕКУЩЕГО СТАТУСА ОДНОГО ИЛИ НЕСКОЛЬКИХ УСТРОЙСТВ, УСТАНОВЛЕННЫХ НА ВАШЕЙ СИСТЕМЕ

Формат команды:

mode *устройство* /STATUS

Для вывода на дисплей статуса всех устройств вашего компьютера, необходимо задать команду MODE в следующем виде:

```
mode
```

Параметр:

устройство - указание имени устройства, статус которого вам необходимо вывести на дисплей.

Ключ	
/STATUS	вывод на дисплей статуса всех переназначенных принтеров. Когда команда MODE задается без этого ключа, она выводит на дисплей статус всех устройств за исключением переназначенных принтеров. Вместо ключа /STATUS допускается использование аббревиатуры /STA.

Пример использования:

Для вывода на дисплей статуса монитора и клавиатуры введите команду:

```
mode con /status
```

ПЕРЕНАЗНАЧЕНИЕ ПАРАЛЛЕЛЬНОГО ПОРТА НА ПОСЛЕДОВАТЕЛЬНЫЙ

Формат команды:

mode LPT*n*:=COM*m*:

Параметры:

LPT*n* - указание номера переназначаемого параллельного порта. Допустимые значения для *n* лежат в пределах от 1 до 3;

COM*m* - указание номера последовательного порта на который будет осуществляться вывод информации, предназначенной для параллельного порта. Значение *m* также лежит в пределах от 1 до 4.

Пример использования:

Допустим, пользователю необходимо, чтобы его компьютер направлял выходную информацию, предназначенную для вывода на принтере с параллельным интерфейсом, на принтер с последовательным интерфейсом. Для решения этой задачи необходимо использовать команду MODE дважды. В первом случае для настройки характеристик последовательного порта, а во втором случае для перенаправления выходной информации с параллельного порта на последовательный, как это сделано в нижеприведенном примере:

```
mode com1 48,e,,,b

mode lpt1=com1
```

В результате выполнения этих команд будет произведена настройка параметров порта COM1 и второй командой MODE параллельный порт LPT1 будет переназначен на порт COM1. Для отмены переназначения портов введите:

```
mode lpt1
```

ПОДГОТОВКА, ВЫБОР, ОБНОВЛЕНИЕ ИЛИ ВЫВОД НА ДИСПЛЕЙ ТАБЛИЦ СИМВОЛОВ (КОДОВЫХ СТРАНИЦ) ДЛЯ ПРИНТЕРА ИЛИ КЛАВИАТУРЫ И МОНИТОРА

Форматы команды:

mode *устройство CODEPAGE PREPARE=yyy... диск:\маршрут\имя_файла*

mode *устройство CODEPAGE SELECT=yyy*

mode *устройство CODEPAGE REFRESH*

mode *устройство CODEPAGE /STATUS*

Параметры:

устройство - указание имени устройства, которое необходимо подготовить к загрузке кодовой страницы. Допускается использовать в этом варианте команды MODE для указания имени устройства следующие обозначения: CON, LPT1, LPT2, LPT3, и PRN.

CODEPAGE PREPARE - подготовка кодовой страницы для указанного устройства. Эта команда используется для того, чтобы подготовить к смене текущую кодовую страницу на указанном устройстве. После подготовки кодовой страницы командой MODE CODEPAGE PREPARE подготовленную кодовую страницы можно сменить командой MODE CODEPAGE SELECT. Вместо аббревиатуры CODEPAGE допускается использование символов CP во всех вариантах команды MODE CODEPAGE, а вместо аббревиатуры PREPARE - символов PREP.

yyy - указание номера кодовой страницы для подготовки или выбора. MS-DOS 6.0 поддерживает кодовые страницы с номерами, которые приведены ниже:

437 - Соединенные Штаты;

850 - Многоязычная;

852 - Словакия;

860 - Португалия;

863 - Канада-Франция;

865 - Норвегия.

диск:\маршрут\имя_файла - полный идентификатор файла, содержащего информацию о кодовых страницах (имеет расширение .CPI), который MS-DOS использует при подготовки кодовых страниц для выбранного устройства.

CODEPAGE SELECT - выбор кодовой страницы, которую вы хотите использовать для указанного устройства. Перед выбором кодовой стра-

ницы она должна подготовлена предварительно командой MODE PREPARE. Вместо аббревиатур CODEPAGE и SELECT допускается использование аббревиатуры SEL.

CODEPAGE REFRESH - перезагрузка и реактивация выбранной кодовой страницы в случае, если возникли проблемы с устройствами компьютера или другие ошибки. Аббревиатуру REFRESH допускается заменять символами REF.

CODEPAGE - вывод на дисплей символов кодовой страницы, если она подготовлена и загружена для указанного устройства.

Ключ

/STATUS	вывод на дисплей символов кодовой страницы, которые подготовлены или выбраны для указанного устройства. Вместо этого ключа допускается использование аббревиатуры /STA. Если же вы попытаетесь ввести команду MODE с именем устройства без ключа /STATUS, утилита MODE выведет подготовленную или выбранную для указанного устройства кодовую страницу.

Пример использования:

```
mode con cp prep=(850,863,860) c:\dos\ega.cpi
```

Шрифты 850, 863, 860 подготавливаются для дисплея EGA или VGA. Шрифты размещаются в файле EGA.CPI, который расположен на диске C: в каталоге \DOS.

Примечания:

MS-DOS включает в свой состав пять файлов с расширением .CPI, которые содержат информацию для следующих устройств:

EGA.CPI	расширенный графический адаптер (EGA) или адаптер компьютера IBM Personal Systen/2;
4201.CPI	IBM Proprinters II и III Model 4201, IBM Proprinters II и III XL Model 4202;
4208.CPI	IBM Proprinter X24E Model 4207, IBM Proprinter XL24E Model 4208;
5202.CPI	IBM Quietwriter III принтер;
LCD.CPI	жидкокристаллический дисплей компьютера IBM PC Convertible.

Для получения дополнительной информации об использовании кодовых страниц, необходимо просмотреть описание команд NLSFUNC и CHCP.

ВЫБОР АКТИВНОГО ДИСПЛЕЙНОГО АДАПТЕРА И РЕЖИМА РАБОТЫ ДИСПЛЕЯ ИЛИ РЕКОНФИГУРАЦИЯ АКТИВНОГО ДИСПЛЕЙНОГО АДАПТЕРА

Формат команды:

mode *дисп_адаптер, сдвиг*, T

mode *дисп_адаптер,n*

mode CON: COLS = *c* LINES = *n*

Параметры:

дисп_адаптер - указание типа дисплейного адаптера. Этот параметр может принимать следующие значения, соответствующие определенному типу адаптера:

- 40 или 80 - устанавливается длина строки в 40 или 80 символов;

- BW40 или BW80 - для цветного графического адаптера CGA (color graphic adapter) включается черно-белый режим отображения информации с числом строк указанных за символами BW;

- CO40 или CO80 - разрешение режима отображения цветной информации на цветном дисплее, с числом линий, соответствующих числам, указанным за символами CO;

- MONO - задание режима работы монохромного монитора, при котором он выводит 80 символов в строке.

сдвиг - задание сдвига изображения на экране монитора CGA влево или вправо. Допустимыми значениями этого параметра являются:

- L - для сдвига влево;

- R - для сдвига вправо;

- T - организация запроса на вывод текстовой строки, позволяющей скорректировать изображение на экране путем его сдвига вправо или влево.

CON: - указание команде MODE, что она должна изменять параметры дисплея.

COLS=*c* - задание числа символов в строке. Допустимыми значениями являются числа 40 и 80.

LINES=*n* - указание числа линий текста одновременно, воспроизводимых на экране дисплея. Допустимыми значениями для этого параметра являются следующие: 25, 43 и 50.

Следует помнить, что не все типы дисплеев поддерживают все три типа значений для *n*. Перед использованием параметра *n* необходимо прочитать в описании вашего компьютера сведения о режимах работы вашего видеоадаптера.

Пример использования:

```
mode bw80
```

После выполнения этой команды для дисплея будет установлен режим вывода текста в черно-белом виде с 80-ю символами в строке.

```
mode,r,t
```

На экран выводится специальная строка для коррекции изображения путем его сдвига вправо.

```
mode con: cols=40 lines=43
```

Устанавливается режим отображения текста с 40 символами в строке и 43 строками.

Примечания:

Для расширения возможностей управления дисплеем необходимо использовать драйвер ANSI.SYS, подключив его к ОС в файле CONFIG.SYS директивой DEVICE.

НАСТРОЙКА ПАРАМЕТРОВ КЛАВИАТУРЫ

Этот вариант команды MODE производит настройку частоты повторения символов и задержки перед повтором символов.

Формат команды:

mode CON: RATE=*r* DELAY=*d*

Параметры:

CON: - указание команде MODE, что она должна настраивать параметры клавиатуры;

RATE=*d* - задание частоты повтора символов после того, как нажатая клавиша клавиатуры удерживается пользователем. Значения параметра *d* лежат в пределах от 1 до 32. Это соответствует примерно частоте повторения от 2 до 30 символов в секунду. По умолчанию частота повтора установлена равной 20 для клавиатур, совместимых с клавиатурой машин IBM PC/AT, и равной 21 для клавиатур, совместимых с клавиатурой компьютера IBM PS/2. При указании в команде MODE частоты повторения символов необходимо указывать и задержку перед их повтором;

DELAY=*d* - указание времени задержки перед повтором символов, после нажатия клавиши. Этот параметр может быть равным следующим числам: 1, 2, 3 и 4. Приведенные значения задержки примерно соответствуют 0.25, 0.50, 0.75 и 1-секундной задержке соответственно.

Пример использования:

mode con: delay=2 rate=3

После выполнения этой команды для клавиатуры устанавливаются следующие характеристики:

- частота повтора символов - 10 символов в секунду;

- задержка перед автоповтором - 0.5 с.

Общие примечания к команде MODE:

В команде MODE при изменении кодовых страниц дисплея не допускается использовать кодовую таблицу, назначенную драйверу DISPLAY.SYS, если он подключен к ОС в файле CONFIG.SYS.

Команду MODE в различных вариантах очень удобно использовать в файле AUTOEXEC.BAT для изменения параметров работы устройств вашего компьютера при загрузке ОС.

MORE

Постраничный вывод содержимого файла. Команда внешняя. Файл MORE.COM.

Стандартным способом использования этой команды является ее применение в операциях перенаправления или в конвейерах.

Команда организует постраничный вывод информации на дисплей. Вывод каждой следующей страницы возможен после нажатия любой клавиши. Утилиту особенно удобно использовать при чтении длинных файлов.

Форматы команды:

more < *диск:\маршрут\имя_файла*
команда | more

Параметры:

диск:\маршрут\имя_файла - указание местоположения и имени файла, содержимое которого необходимо вывести на дисплей.

команда - указание команды, выходная информация которой будет подана на вход команды MORE.

Примеры использования:

Допустим, пользователю необходимо прочитать содержимое длинного файла с именем REPORT.TXT. Чтобы сделать, это нужно использовать команду MORE так, как это показано ниже:

more < report.txt

```
type report.txt | more
```

Оба приведенных примера выполняют одну и ту же операцию. Файл REPORT.TXT будет постранично выведен на экран. После вывода на дисплей одной страницы текста внизу появится следующее сообщение:

```
More
```

Для просмотра следующей страницы текста необходимо нажать любую клавишу и просмотр будет продолжен.

Примечания:

- При использовании символа перенаправления "<" необходимо указывать после команды MODE *имя_файла*, который будет прочитан. При использовании символа конвейеризации "|" команда MORE может совместно использоваться со следующими командами DOS: DIR, SORT и TYPE.
- Для более быстрой работы DOS в операциях конвейеризации необходимо установить в файле AUTOEXEC.BAT окружение для переменной TEMP.
- Для получения дополнительных сведений о командах DIR, SORT и TYPE необходимо просмотреть их описания.

(!) MOVE

Перемещение одного или нескольких файлов в указанное на диске место. Команда MOVE также производит переименование каталогов. Команда внешняя. Файл MOVE.EXE.

Формат команды:

move *диск1:\маршрут1\имя_файла1, диск2:\маршрут2\имя_файла2,... диск:\маршрут*

Параметры:

диск1:\маршрут1\имя_файла1, диск2:\маршрут2\имя_файла2,... - указание полного идентификатора 1-го, 2-го, n-го файла или группы файлов, подлежащих перемещению в другое место на диске;

диск:\маршрут - указание нового месторасположения файлов или нового имени каталога. Параметр может включать в себя букву диска или имя каталога, а также их комбинацию. Если производится перемещение одного файла, то в параметре *диск:\маршрут* можно указывать новое имя файла, которое он получит после его перемещения. Если файл в каталоге назначения уже существует, то перемещаемый файл с таким же именем и расширением будет записан поверх уже существующего файла. Если же перемещаются несколько файлов, то в пара-

метре *диск:\маршрут* необходимо использовать только букву диска и имя каталога.

Примеры использования:

Для перемещения файлов PRIVET.DOC и OTVET.DOC из текущего каталога в каталог диска C:\LETTERS введите команду:

```
move privet.doc, otvet.doc c:\letters
```

Если вам необходимо переименовать файл с именем PETR.DOC из каталога LETTERS диска C: в файл с именем VICTOR.DOC необходимо использовать команду MOVE в следующем варианте:

```
move petr.doc c:\letters\victor.doc
```

Переименование каталога LETTERS в каталог NOTES можно провести при помощи следующей команды:

```
move c:\letters c:\notes
```

Примечания:

- При попытке перемещения нескольких файлов с одновременным их переименованием DOS выведет на дисплей следующее сообщение:

  ```
  Cannot move multiple files to a single file
  (Не могу переместить несколько файлов в один файл)
  ```

- При изменении имени каталога командой MOVE вы не сможете расположить каталог со старым именем в другом месте дерева каталогов. Для примера, переименуем каталог с именем TOOLS на диске C: в каталог SYS командой:

  ```
  move c:\tools c:\sys
  ```

  Если вы попытаетесь найти потом каталог с именем TOOLS в структуре дерева каталогов, то вы его не обнаружите.

  Приведенный ниже пример показывает неправильный формат задания команды MOVE для переименования каталогов. Если вы попытаетесь следующей командой:

  ```
  move c:\new c:\dos\new
  ```

  переименовать каталог NEW в подчиненный каталогу DOS каталог с тем же именем NEW, DOS выведет сообщение об ошибке.

- Команда MOVE возвращает DOS следующие коды своего завершения:

0	проведено успешное перемещение всех файлов;
1	при перемещении одного или нескольких файлов обнаружена ошибка.

Коды завершения в дальнейшем могут использоваться другими прикладными программами или пакетными файлами, например, в параметре ERRORLEVEL команды IF.

(!) MSAV

Сканирование вашего компьютера на вирусы. Команда внешняя. Файл MSAV.EXE. Эта утилита позволяет обнаружить на машине присутствие компьютерных вирусов, уничтожить вирусы и провести лечение, зараженных вирусами файлов. Утилита позволяет обнаруживать 1234 известных к моменту ее появления вируса. Команда MSAV после ее вода приводит к появления программного меню, внешний вид которого представлен на рисунке 8.8.

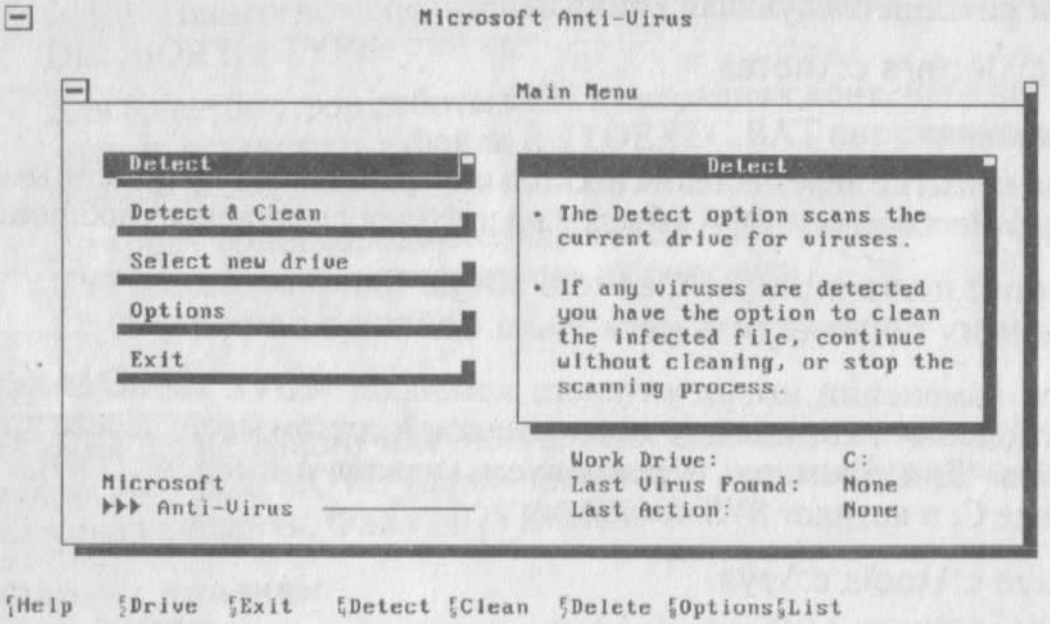


Рис. 8.8

По своему внешнему виду и принципам управления утилитой MSAV (Microsoft Anti - Virus) очень похож на другую известную программу Central Point Anti - Virus фирмы Central Point Software Inc. Это объясняется тем, что программа MSAV разработана для MS-DOS 6.0 фирмой-изготовителем Central Point Anti - Virus.

В центре экрана расположено главное меню программы, разделенное на 2 части. Справа отображается текущее состояние утилиты, а слева имеются пункты главного меню, позволяющие выбрать необходимый режим работы MSAV. Главное меню программы содержит следующие пункты:

- "Detect" - обнаружение вирусов в памяти и на логических дисках вашего компьютера. Перед выбором этого пункта следует выбрать логический диск для проверки. Если этого не сделано, то при первом за сеанс работы запуске MSAV, будет проверена оперативная память компьютера, а затем текущий логический диск. Справа в главном меню будут отображаться данные о проверке памяти, а затем и логических дисков, включая наименование каталогов и проверяемых файлов;

- "Detect & Clean" - обнаружение вирусов с их уничтожением. В этом пункте действия и выходная информация MSAV аналогичны предыдущему пункту, за исключением того, что MSAV не будет запрашивать ответа пользователя при обнаружении подозрительных действий на дисках и в памяти;

- "Select New Drive" - выбрав этот пункт главного меню пользователь может при помощи "мыши", либо клавиатуры выбрать логический диск для проверки, в строке поверх главного меню;

- "Options" - задание параметров работы MSAV. При выборе этого пункта пользователь увидит еще одно окно "Options Setting". Здесь задаются следующие установки:

- "Verify Integrity" - общая проверка компьютера;
- "Create New Checksums" - создавать новые контрольные суммы;
- "Create New Checksums on Floppy" - создавать новые контрольные суммы на гибких дисках;
- "Disable Alarm Sound" - запретить подачу звукового сигнала при обнаружении подозрительных действий;
- "Create Backup" - создавать резервные копии;
- "Create Report" - создавать файл MSAV.TXT, содержащий отчет о проверке дисков;
- "Prompt While Detect" - выдавать подсказку при проверке;
- "Anti - Stealth" - организовывать работу по алгоритму, позволяющему бороться со скрытыми вирусами;
- "Check All files" - проверять все файлы подряд, а не только те, которые проверяются по умолчанию.

- "Exit" - выход из программы MSAV. При выходе из программы MSAV выдает запрос на сохранение своих текущих установок. Ответив "Ok", вы сохраните их, ответив "Cancel", - отмените текущие установки MSAV, и по запуску утилита установит сама параметры по умолчанию.

Управление программными меню MSAV может осуществляться как с помощью клавиатуры, так и с помощью "мыши". Для выбора пункта главного меню достаточно нажать клавишу, соответствующую подсвеченному символу в пункте главного меню. Перемещение по пунктам меню программы осуществляется с помощью клавиш управления курсором и клавишей Tab.

Кроме этого, управление программой возможно и с помощью специальных клавиш F1, F2, F3, F4, F5, F7, F8, F9. Функции, закрепленные за

указанными клавишами, приведены в строке, помещенной внизу экрана программы.

F1 вызывает вывод на дисплей справочной информации о программе MSAV.

F2 позволяет выбрать логический диск для проверки.

F3 осуществляет выход из программы Microsoft Anti-Virus.

F4 производит проверку "Detect".

F5 производит проверку и очистку "Detect & Clean".

F7 позволяет очистить диск от файлов CHKLIST.MS, содержащих контрольные суммы файлов, проверенных с помощью MSAV.

F8 открывает окно "Options Setting".

F9 выводит перед пользователем список вирусов, которые может обнаружить и уничтожить утилита MSAV. При выборе этого пункта программного меню раскрывается окно с именем "Virus List" в котором можно:

- быстро найти информацию о вирусе, введя его имя в подсвеченное поле внизу экрана справа;
- найти информацию о следующем вирусе по их списку - "Find Next";
- получить краткую информацию о вирусе - "Info";
- распечатать сведения о вирусах на принтере - "Print".

Перемещение по списку вирусов осуществляется с помощью клавиш управления курсором в таблице посреди экрана. Таблица имеет следующие столбцы:

"Name" - имя вируса;

"Type" - тип вируса (файловый - File, бутовый - Boot, троянский - Trojan);

"Size" - длина вируса в байтах;

"#" - номер типа вируса.

Так как все возможности программы MSAV не могут быть реализованы через ее меню, DOS предоставляет возможность управления параметрами MSAV и с помощью стандартного для DOS вызова MSAV с ключами и параметрами из командной строки.

Формат команды:

msav *диск:* /S|/C /R /A|/L /N /P /F /VIDEO

Параметр:

диск: - указание диска, который должен быть проверен MSAV на вирусы. Если этот параметр опущен, MSAV проверяет текущий диск.

Ключи

/S	сканирование указанного диска без уничтожения обнаруженных MSAV вирусов;
/C	сканирование указанного диска с уничтожением обнаруженных MSAV вирусов;
/R	создание файла MSAV.RPT содержащего число проверенных MSAV файлов, число обнаруженных вирусов и число вирусов, которые уничтожены утилитой MSAV. По умолчанию MSAV не создает отчета. После того, как проверка диска будет завершена, файл MSAV.RPT будет записан в корневой каталог;
/A	проверка всех дисков компьютера, кроме дисков A: и B:;
/L	проверка всех локальных дисков, исключая сетевые диски;
/N	вывод на дисплей содержимого файла MSAV.TXT, если он уже существует и располагается в том же каталоге, что и файл MSAV.EXE. При использовании этого ключа MSAV сканирует либо тот диск (диски), который указал пользователь, либо текущий диск, и при обнаружении вируса включает в файл код завершения 86, который приводит к появлению на экране сообщения об ошибке. Этот код завершения можно использовать в прикладных программах и в командных файлах (см. главу "Командные файлы").
/P	использование командной строки вместо графического интерфейса;
/F	отключение вывода на дисплей имен файлов, которые подвергаются проверке; Этот ключ чаще всего используется совместно с ключами /N или /P;
/VIDEO	ключ, задающий режимы работы дисплея. Вместо этого ключа необходимо использовать ключи, обозначения которых и решаемые ими задачи приведены ниже: /25 - установка режима работы дисплея в 25 строк информации. Этот ключ действует по умолчанию; /28 - установка режима работы дисплея в 28 строк. Этот ключ действует только для видеоадаптеров VGA и выше; /43 - вывод на дисплей 43 строк информации. Применяйте этот ключ для видеоадаптеров EGA и VGA; /50 - вывод на дисплей 50 строк информации. Ключ действителен для видеоадаптеров VGA; /60 - вывод на дисплей 60 строк информации. Используйте этот ключ только для видеоадаптеров Video 7;

/IN	запуск MSAV в цветном режиме, если цветной видеоадаптер не обнаружен;
/BW	запуск MSAV в черно-белом режиме отображения информации;
/MONO	запуск MSAV на компьютере, оснащенном монохромным монитором;
/LCD	запуск MSAV с использованием жидкокристаллического (LCD) дисплея;
/FF	ликвидация эффекта "снег" на компьютерах, оснащенных дисплеем CGA. Используйте этот ключ для повышения качества изображения на экране дисплея;
/BF	использование BIOS компьютера для работы программы с дисплеем;
/NF	запрещение использования альтернативных шрифтов;
/BT	разрешение использования графической "мыши" при работе в среде Windows;
/NGM	запуск утилиты MSAV с использованием характеристик "мыши", установленных по умолчанию, включая графические символы;
/LE	замена местами правой и левой кнопок "мыши";
/PS2	сброс мыши, если курсор исчез или запрещен. Ключ используется для компьютеров IBM PS/2 или совместимых с ними.

Примеры использования:

Для запуска утилиты MSAV на компьютере с дисплеем VGA, в режиме отображения 28 строк, и проверки всех дисков машины, за исключением дисков A: и B:, с созданием файла MSAV.RPT, введите команду:

```
msav /28 /A /R
```

Если же вам необходимо запустить программу MSAV на компьютере с дисплеем CGA с ликвидацией эффекта "снег", с уничтожением всех обнаруженных вирусов, следует использовать команду:

```
msav /FF /C
```

Примечания:

- В процессе работы Microsoft Anti-Virus создает файлы CHKLIST.MS в каждом каталоге, где производилась проверка на вирусы. При первом запуске MSAV он записывает в этот файл контрольные суммы, соответствующие каждому проверенному файлу. При последующих запусках MSAV сравнивает контрольные суммы файлов, записанные в CHKLIST.MS с контрольными суммами, вычис-

ленными в процессе работы. При несоответствии контрольных сумм пользователю выдается сообщение о заражении файла вирусами.

- К наиболее характерным симптомам появления на компьютере вирусов MSAV относит следующие:

  - повреждение программ.

  - появление неизвестных резидентных программ.

  - появление конфликтов при работе драйверов устройств.

  Эти проблемы могут также возникать и при неисправности оборудования компьютера или программ, но для того, чтобы убедиться, что проблемы возникли не из за вирусов, необходимо запустить MSAV.

- При выборе установки "Check All Files" MSAV проверяет все файлы на дисках, включая файлы с расширениями: .386, .APP, .BIN, .CMD, .DOM, .DLL, .DRV, .EXE, .FON, .ICO, .OV*, .PGM, .PIF, .PRG, .SYS. Файлы с указанными расширениями всегда проверяются MSAV, независимо от того, выбрана установка "Check All Files" или нет.

- При запуске утилиты MSAV она использует для своей инициализации файл MSAV.INI, содержащий информации о последних установках программы. Для определения местоположения файла MSAV.INI программа сначала анализирует местоположение файла MSAV.INI, просматривая окружение переменной MSDOSDATA. Если это окружение не установлено, поиск файла MSAV.INI осуществляется в том же каталоге, в котором находится файл MSAV.EXE. Если и после этого файл MSAV.INI не будет найден, то он будет создан автоматически утилитой MSAV в том же каталоге, где находится выполняемый модуль MSAV.EXE. Для ускорения запуска MSAV необходимо командой SET установить окружение для переменной MSDOSDATA, как это сделано в приведенном ниже примере:

  ```
  set msdosdata = c:\dos
  ```

При работе с Microsoft Anti-Virus все изменения будут записываться в файл MSAV.INI, расположенный на диске C: в каталоге DOS.

(!) MSBACKUP

Резервное копирование или восстановление одного или нескольких файлов с одного диска на другой. Используя эту утилиту пользователь может провести резервное копирование всех файлов на диске или только файлов, измененных со времени последнего резервного копирова-

ния. Систематическое резервное копирование с помощью утилиты MSBACKUP поможет вам сохранить свои данные на случай непредвиденных обстоятельств, которые могут произойти с вашим жестким диском.

Команда внешняя. Файл MSBACKUP.EXE.

Утилита после загрузки выводит на экран диалоговое меню (Рис 8.9), из которого осуществляется все управление программы Microsoft Backup.

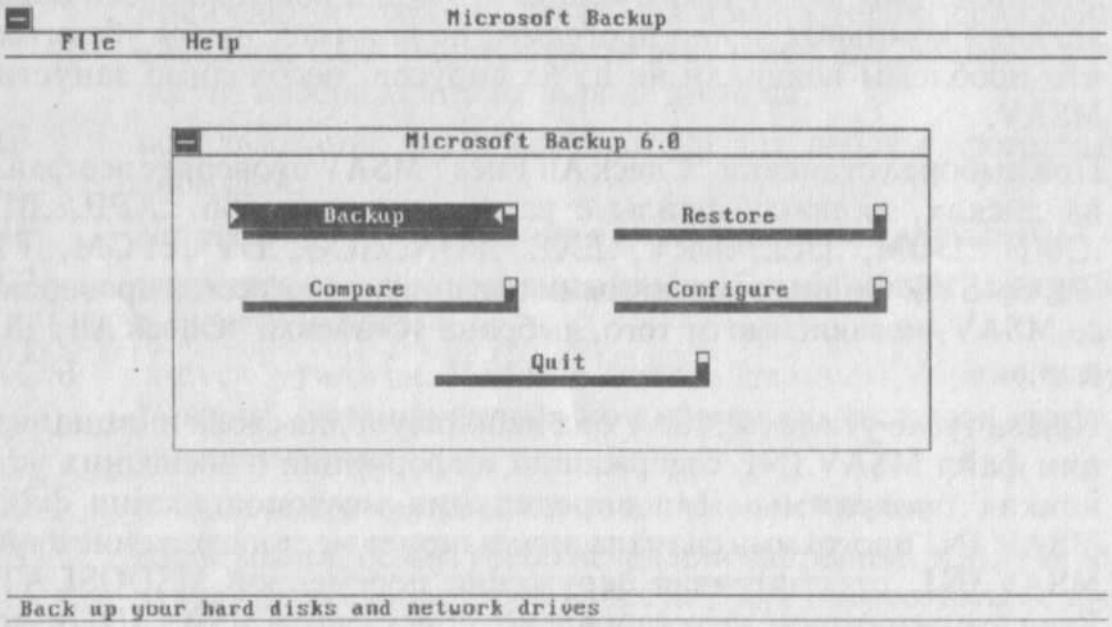


Рис. 8.9

В центре экрана содержится диалоговое меню, содержащее пункты:

- "Backup" - резервное копирование данных;
- "Compare" - сравнение файлов, каталогов, дисков;
- "Restore" - восстановление данных с резервных дисков;
- "Configure" - установка конфигурации Microsoft Backup;
- "Exit" - выход из программы MSBACKUP.

О назначении пунктов главного диалогового окна рассказывается ниже.

В верхней части экрана располагается строка меню, в которой имеется 2 пункта: "File" и "Help".

В пункте "File" вы можете выбрать пункты:

- "Open Setup..." - открытие файла, содержащего установки для программы Microsoft Backup. Открыв этот файл, вы увидите установки программы MSBACKUP, и после этого можете их изменить;

- "Save Setup" - запись установок MSBACKUP в файл, содержащей программные установки;
- "Save Setup As..." - запись установок MSBACKUP в файл с именем, отличным от предыдущего файла установок программы. По умолчанию таким файлом является файл с именем DEFAULT.SET;
- "Delete Setup" - удаление предыдущих установок программы MSBACKUP. При запуске программы с файлом DEFAULT.SET, не содержащим установок MSBACKUP, или без него, утилита установит свои параметры по умолчанию. Если файл DEFAULT.SET не существует или не найден, MSBACKUP создает его в процессе своей загрузки;
- "Print" - печать установок программы MSBACKUP на принтере;
- "Printer Setup..." - выбор используемого для связи с принтером порта и установка характеристик печати самого принтера;
- "Exit" - выход из программы Microsoft Backup.

Пункт "Help" в верхнем меню программы содержит пункты:

- "Index" - поиск помощи по индексу;
- "Keyboard" - справочная информация об управлении Microsoft Backup с помощью клавиатуры;
- "Basic Skills" - информация об основных приемах работы с MSBACKUP;
- "Backup" - этот пункт содержит информацию о том как, проводить резервное копирование;
- "Compare" - пункт содержит справочную информацию о проведении операций сравнения файлов, каталогов и дисков;
- "Restore" - прочитав этот пункт пользователь получит сведения о том, как правильно проводить восстановление файлов с резервных дискет;
- "Configure" - в этом пункте содержится информация о настройке параметров программы MSBACKUP;
- "Glossary" - этот пункт содержит сведения о терминах и определениях, связанных с операциями резервного копирования и восстановления файлов;
- "Using Help" - в этом пункте содержатся сведения о том как, пользоваться системой помощи утилиты MSBACKUP;
- "About Help" - общая информация о программе MSBACKUP и фирме - изготовителе. Использование главного диалогового окна является основным методом работы пользователя с MSBACKUP.

При выборе пункта "Backup" перед пользователем откроется новое диалоговое окно, позволяющее настроить параметры резервного копирования. В появившемся диалоговом меню имеются следующие пункты:

- "Setup File" - задание имени файла, в котором сохраняются установки программы MSBACKUP. По умолчанию файл настройки имеет имя DEFAULT.SET;
- "Backup From" - выбор диска и(или) каталога, файлы которого подлежат резервному копированию;
- "Backup To" - выбор диска и каталога куда будет производиться запись файлов резервного копирования. Здесь пользователь может выбрать следующие варианты:
  - "Original locations" - оригинальное место на диске;
  - "Other drives" - другие диски;
  - "Others directories" - другие каталоги;
- "Select Files" - при выборе этого пункта в окне "Backup", пользователь может выбрать файл или группу файлов для которого он хочет сделать резервную копию. Справа будет отображаться структура каталогов выбранного диска, слева содержимое выбранного каталога. Если при запуске MSBACKUP диск не выбран, то считается, что работа должна быть организована с текущим диском. Под изображением

структуры каталогов и списка файлов выбранного каталога отображаются еще несколько пунктов меню "Select Files":

- "Include" - позволяет включить в список файлы для резервного копирования. Кроме этого выбора файлов по дискам и каталогам в список файлов для резервного копирования можно включить, все подчиненные выбранному каталогу, подкаталоги - пункт "Include All SubDirectories". Возможно также редактирование выбранных для резервного копирования файлов - пункт "Edit Include\Exclude Files";
- "Exclude" - исключение из списка выбранных файлов, отдельных файлов, используя специальные установки в открывшемся после выбора пункта "Exclude" окне. Возможно исключение файлов по следующим признакам:
  - По дате их создания.
  - Файлов, защищенных от копирования - "Exclude Copy Protected Files".
  - Файлов, имеющих атрибут "только для чтения" - "Exclude Read-Only Files".
  - Системных файлов - "Exclude System Files".
  - "Скрытых" файлов - "Exclude Hidden Files".
- "Special" - в этом пункте возможно исключение специальных файлов. Работа с этим пунктом меню "Select Files" аналогична работе с пунктом "Exclude" этого же окна;
- "Display" - этот пункт позволяет организовать сортировку файлов при выводе их списка на экран. Сортировка возможна по следующим признакам:
  - Не сортировать ("Unsorted").
  - По имени ("Name").
  - По расширению ("Extension").
  - По длине ("Size").
  - По дате создания ("Date").
  - По атрибутам файла ("Attributes").

  Кроме того, пользователь имеет возможность пометить группу файлов - "Group Selected Files", задав групповой шаблон в другом пункте "File Filter". По умолчанию принимается шаблон "*.*";
- Пункт меню "Backup" - "Backup Type" позволяет выбрать тип резервного копирования. Имеется возможность выбрать три типа резервного копирования:
  - "Full" - полное копирование всех файлов на диске.
  - "Incremental" - добавочное копирование только тех файлов, которые были изменены со времени последнего резервного копирования.
  - "Differental" - выборочное копирование только тех файлов, которые выбрал пользователь.
- Выбор пользователем пункта "Start Backup" предписывает MSBACKUP начать резервное копирование указанных ранее резервных файлов.
- "Options" - этот пункт меню "Backup" служит для задания программе MSBACKUP установок при проведении процесса резервного копирования файлов. Пользователю доступны следующие установки:
  - "Verify Backup Data (Read and Compare)" - проверка данных резервного копирования (Чтение и сравнение данных);
  - "Compress Backup Data" - сжатие данных при резервном копировании;
  - "Password Protect Backup Sets" - защита паролем установок программы Microsoft Backup;
  - "Prompt before overwriting used diskettes" - выдача на экран подсказки при записи информации поверх старой на дискету;
  - "Always format diskettes" - форматирование дискет при обнаружении неотформатированной дискеты;
  - "Use error correction on diskettes" - коррекция ошибок при работе с дискетами;
  - "Keep old Backup catalogs" - сохранение старого каталога Backup;
  - "Audible prompt (Beep)" - "слышимая" подсказка-звонок;
  - "Quit after Backup" - выход из окна после резервного копирования.

Пункт главного окна программы "Compare" позволяет осуществить сравнение дисков, каталогов и файлов. Этот пункт главного диалогового окна открывает еще одно окно, включающее в свою очередь следующие разделы:

- "Backup Set Catalog" - выбор каталога для программы MSBACKUP;
- "Compare From" - выбор диска и каталога, с которого будет произведено резервное копирование. В этом разделе можно выбрать два возможных варианта:
  - "Floppy disks" - гибкие диски.
  - "MS-DOS Drive and Path" - диски MS-DOS и пути.
- "Compare To" - выбор места расположения файлов резервного копирования. Пользователю предлагается выбрать:
  - "Original locations" - оригинальное место на диске.
  - "Other drives" - другие диски.
  - "Others directories" - другие каталоги.
- Пункт окна "Compare Files" - содержит информацию о файлах, выбранных для сравнения.
- Выбор пользователем пункта "Start Compare" предписывает MSBACKUP начать сравнение указанных ранее резервных файлов.

В пункте меню "Options" можно установить:

- "Слышимую" подсказку - "Audiable prompt (Beep).
- Задать выход из окна "Compare" после окончания сравнения.

В пункте "Catalog" окна "Compare" пользователь может выбрать диски и каталоги, файлы которых будут сравниваться. В раскрывшемся окне появятся следующие пункты:

- "Load" - загрузка файлов в память;
- "Retrieve..." - сравнение дисков и каталогов с безвозвратным уничтожением информации. При выборе этого пункта необходимо выбрать тип диска на котором будет проводиться сравнение;
- "Rebuild..." - восстановление заново сравниваемых файлов и каталогов. Необходимо выбрать перед сравнением диск и каталог;
- "Delete" - удаление информации программы MSBACKUP с дисков.

Пункт главного окна "Restore" программы Microsoft Backup позволяет восстановить данные резервного копирования, созданные при помощи пункта "Backup" утилиты MSAV. Он содержит следующие пункты:

- "Backup Set Catalog" - выбор диска и(или) каталога, содержащего файл с установками программы MSAV;
- "Restore From" - выбор диска и(или) каталога, с которого будет производиться восстановление файлов резервного копирования;
- "Backup To" - выбор диска и каталога, куда будет производится запись файлов резервного копирования. Пользователь может выбрать следующие варианты:
  - "Original locations" - оригинальное место на диске.
  - "Other drives" - другие диски.
  - "Others directories" - другие каталоги.
- Пункт "Select Files" аналогичен по своей структуре и способам выбора файлов пункту с тем же названием в окне "Backup".
- Выбор пользователем пункта "Start Restore" предписывает MSBACKUP начать восстановление указанных ранее резервных файлов.

Пункты "Options" и "Catalog" аналогичны по своему построению этим же пунктам в окне "Compare".

В пункте "Configure" главного окна можно установить конфигурацию компьютера по своему усмотрению.

Пункт "Video and Mouse" позволяет настроить параметры дисплея и "мыши" по своему усмотрению. Раскрывающееся при выборе этого пункта новое окно содержит пункты:

Для настройки дисплея "Display Opions" они следующие:

- "Screen Colors". Здесь возможны следующие варианты:
- "Laptop" - настройка для дисплея LCD портативных компьютеров;
- "Monochrome" - работа с монохромным монитором;
- "CGA Colors" - настройка на цвета CGA;
- "EGA/VGA Colors 1" - выбор палитры цветов для дисплеев EGA/VGA номер 1;
- "EGA/VGA Colors 2" - выбор палитры цветов для дисплеев EGA/VGA номер 2;
- "Display Lines" - установка количества линий для отображения информации. Выбрав этот пункт, можно задать следующие установки:
  - "Default lines" - установка количества линий по умолчанию;
  - "25 lines" - режим отображения информации в 25 строк;
  - "28 lines" - режим отображения информации в 28 строк;
  - "43 lines" - режим отображения информации в 43 строки;
  - "50 lines" - режим отображения информации в 50 строк.

В пункте окна "Display Options" доступны следующие установки:

- "Standard" - стандартный режим работы дисплея;
- "Graphical Controls" - графический контроль;
- "Graphical Controls and Mouse" - графический контроль и "мышь";
- "Graphical controls, Mouse and Dialogs" - графический контроль, "мышь" и графический диалог.

Для настройки параметров "мыши" служит раздел "Mouse Options". Здесь можно настроить скорость нажатия клавиш "мыши" - "Double-Click". Возможны следующие варианты установки этого параметра:

- "Slow" - медленная скорость;
- "Medium" - средняя скорость;
- "Fast" - быстрая скорость.

Для настройки чувствительности "мыши" - "Sensivity" имеются варианты:

- "Default" - настройка по умолчанию;
- "Low" - низкая чувствительность;
- "Medium" - средняя чувствительность;
- "High" - высокая чувствительность.

Скорость движения "мыши" можно изменять в пункте "Acceleration", где имеются следующие варианты выбора:

- "Default" - настройка по умолчанию;
- "Low" - низкая чувствительность;
- "Medium" - средняя чувствительность;
- "High" - высокая чувствительность.

Кроме того, в нижней части экрана для дисплея и "мыши", имеются следующие варианты настройки:

- "Reduce Display Speed" - уменьшение скорости отображения информации;
- "Expanding Dialogs" - расширенный диалог MSBACKUP с пользователем;
- "Left - Handed Mouse" - использование мыши левой рукой;
- "Hard Mouse Reset" - жесткий сброс "мыши".

Пункт "Backup Device" в окне "Configure" позволяет выбрать тип гибкого диска для проведения операций резервного копирования.

В пункте "Compability Test" имеется возможность проверить на совместимость различные дискеты. При выполнении этого пункта MSBACKUP позволяет провести мини-тест на резервное копирование и восстановление данных с гибких дисков. Раскрывающееся при этом окно содержит внизу пункты, позволяющие:

- "State Test" - выбрать тест;
- "Skip Test" - пропустить тест;
- "Cancel" - отменить проверку.

Пункт "Save" окна "Configure" позволяет осуществить запись установок программы MSBACKUP.

Пункт главного окна "Exit" осуществляет выход из программы MSBACKUP.

Управление программой MSBACKUP осуществляется как при помощи клавиш, так и при помощи манипулятора типа "мышь". Перемещение по пунктам меню возможно как с помощью клавиш управления курсором, так и клавишей Tab, а также нажатием клавиши, соответствующей подсвеченному символу в пунктах окон программы.

Если вы пользуетесь "мышью", достаточно подвести указатель "мыши" к необходимому пункту и нажать левую (правую) кнопку "мыши" для его выбора.

Те пункты меню в окнах, которые недоступны в текущий момент, подсвечиваются неярким цветом и отображаются прописными буквами. Управление MSBACKUP возможно также из командной строки DOS.

Формат команды:

msbackup *имя_уст_файла* /BW|/LCD|/MDA

Параметр:

имя_уст_файла - указание имени файла, содержащего информацию о стартовых установках утилиты MSBACKUP. Microsoft Backup создает файл установки, когда записывает установки программы и выбора файлов. Файлы установки для программы MSBACKUP имеют расширение .SET.

Ключи

/BW	запуск программы MSBACKUP в черно-белом режиме отображения информации;
/LCD	запуск утилиты на компьютерах с дисплеем LCD;
/MDA	запуск MSBACKUP на компьютере с монохромным дисплеем.

Пример использования:

Допустим, вы сохранили текущие установки программы MSBACKUP в файле с именем PRIMER.SET и хотите его использовать для проведения операции резервного копирования указанных файлов. Чтобы использовать этот файл для настройки MSBACKUP в процессе его запуска, введите команду:

```
msbackup primer
```

Примечания:

Утилита MSBACKUP может запускаться только с жесткого диска, эта утилита не может быть запущена с гибких дисков.

В процессе работы MSBACKUP создает свои каталоги, которые содержат информацию о файлах, подлежащих резервному копированию. Когда вам необходимо восстановить один или несколько файлов, вы должны загрузить каталог Backup и затем выбрать файлы для копирования или восстановления. Каталог программы BACKUP содержит информацию по следующим темам:

- Структуру каталогов диска после резервного копирования.
- Имена, длины и атрибуты выбранных каталогов и файлов.
- Общее число файлов.
- Общий размер резервного копирования.
- Имя используемого установочного файла.
- Дата создания последней резервной копии.

MSBACKUP присваивает каждому файлу каталога уникальное имя. Каждый символ в имени каталога имеет определенное значение. Например, рассмотрим файл с именем CE31122A.FUL. Читая справа налево, можно получить подробную информацию о созданном MSBACKUP резервном файле:

Символ	Его значение
C	Буква, обозначающая первый диск резервного копирования
E	Буква, обозначающая последний диск резервного копирования. Если последним диском был первый диск копирования, то эта буква будет совпадать с первой буквой имени файла
3	Последняя цифра года системной даты. В приведенном примере цифра 3 указывает на то, что копирование проведено в 1993 году
11	Месяц, в который создан файл резервного копирования
22	День месяца, в который проведено резервное копирование
A	Эта позиция показывает номер копии за указанный ранее день. Если в среде Microsoft Backup задействован пункт "Keep Old Backup Catalogs", то программа MSBACKUP добавляет к имени файла буквы от A до Z для указании номера резервной копии за указанный день. Буква A соответствует первой копии за день, B - второй, C - третьей и так далее. Если установка "Keep Old Catalogs" не используется, то значение этого параметра находится в пределах от A до B.
FUL	Индикация типа резервного копирования. В нашем случае было произведено полное копирование (Full - .FUL). Выборочное копирование - .DIF, добавочное копирование - .INC.

Каждый раз при проведении полного резервного копирования с использованием своего файла установки, MSBACKUP создает главный каталог. Главный каталог содержит записи о всех каталогах, включенных программой MSBACKUP в цикл резервного копирования. Каждый раз при запуске программы MSBACKUP для полного копирования файлов и начале нового цикла резервного копирования файлов она создает новый главный каталог.

Главный каталог используется в дальнейшем для организации цикла восстановления данных резервного копирования. Когда вы загружаете в память главный каталог, каталоги со всеми данными резервного копирования, созданные во время цикла копирования, автоматически будут уничтожены. После этого последняя версия каждого файла резервного копирования будет автоматически восстановлена.

При проведении копирования файлов утилита MSBACKUP создает одну копию каталога файлов резервного копирования на жестком диске, и вторую копию на диске или на сетевом диске, включенном в установки программы MSBACKUP.

Так как программа MSBACKUP требует для своей работы значительные ресурсы памяти, при ее загрузке DOS может вывести сообщение об ошибке:

```
Insufficient memory
```

Для работы с MSBACKUP ваш компьютер должен иметь 512 и более килобайт оперативной памяти. При появлении сообщений об ошибке необходимо выйти из программы MSBACKUP, выгрузить из памяти все резидентные программы и вновь загрузить MSBACKUP. Если и указанные меры не помогут организовать нормальную работу с программой, необходимо отключить сжатие данных резервного копирования - "Compress Backup Data" в диалоговом окне "Backup".

Для ускорения запуска MSBACKUP необходимо командой SET установить окружение для переменной MSDOSDATA, как это сделано в приведенном ниже примере:

```
set msdosdata = c:\dos
```

Программа MSBACKUP использует окружение переменной MSDATA для поиска файлов, содержащих установки Microsoft Backup, настройку конфигурации программы и каталоги MSBACKUP. Если файлы MSBACKUP не найдены, в указанных окружением MSDOSDATA каталогах и дисках, утилита ищет свои файлы том же каталоге, где находится файл MSBACKUP.EXE. В том случае, если файлы установки и настройки программы не найдены, программа MSBACKUP сама создаст их.

Окружение переменной MSDOSDATA обычно используется для указания места расположения программных файлов MSBACKUP в каталогах, отличных от того, где находится файл MSBACKUP.EXE. Допустим, программные файлы MSBACKUP находятся в каталоге \DOS диска C:, а сам файл MSBACKUP - на сетевом диске Z:. В этом случае необходимо включить в файл AUTOEXEC.BAT следующую строку:

```
set msdosdata = c:\backup
```

Если после этого набрать команду:

```
z:\msbackup
```

то программа MSBACKUP будет запущена сетевого диска с параметрами, задаваемыми программными файлами, расположенными в каталоге BACKUP диска C:. При внесении изменений в настройку MSBACKUP, все изменения будут записаны в файлы каталога BACKUP диска C:.

(!) MSCDEX

Обеспечение работы DOS с дисками CD-ROM (лазерными компакт-дисками). Команда внешняя. Файл MSCDEX.EXE. Команда MSCDEX может быть включена, как в состав вашего файла AUTOEXEC.BAT, так и задана с использованием командной строки DOS. Работа с коман-

дой MSCDEX возможна только в том случае, если драйвер установленного на компьютере CD-ROM диска подключен к ОС директивой DEVICE в файле CONFIG.SYS. Команда MSCDEX не может быть вызвана на выполнение при работе в среде Microsoft Windows.

Формат команды:

mscdex /D:*драйвер1* /D:*драйвер2*.. /E /K /S /V /L:*буква* /M:*номер*

Параметры:

/D:*драйвер1* /D:*драйвер2*... - задание признаков драйвера для первого (второго и т.д.) драйвера дисков CD-ROM. Параметры *драйвер1*, *драйвер2* и т.д. должны совпадать с параметрами, указанными после ключа /D в файле CONFIG.SYS при описании параметров подключаемых к ОС драйверов лазерных компакт - дисков.

Ключи

/E	указание драйверу дисков CD-ROM, что он должен использовать расширенную память (если она доступна) для размещения секторных буферов.
/K	этот ключ предписывает MS-DOS реорганизовать тома диска CD-ROM, закодированные в формате "Kanji". По умолчанию MS-DOS не реорганизует тома CD-ROM дисков в формате "Kanji".
/S	разрешение разделения дисков в ЛВС MS-NET или в сети "Windows for Workgroups servers".
/V	вывод на дисплей утилитой MSCDEX отчета об использовании памяти перед ее запуском.
/L:*буква*	указание буквы алфавита, которую необходимо присвоить первому CD-ROM диску. Если ваш компьютер оснащен более чем одним CD-ROM диском, MS-DOS присваивает компакт-дискам буквы, следующие по алфавиту за буквами, обозначающими последний доступный ОС логический диск.
/M:*число*	указание числа секторных буферов.

Примеры использования:

Для загрузки драйвера и разрешения использования DOS одного диска CD-ROM необходимо включить в состав файлов CONFIG.SYS и AUTOEXEC.BAT команды и директивы:

- в файл CONFIG.SYS необходимо ввести директиву:

```
device = c:\sys\cdromdrv.sys /D:mscd000
```

Эта директива объявляет подключение драйвера CD-ROM диска CDROMDRV.SYS, который объявляет, что подключаемый диск имеет признак MSCD000;

- в файл AUTOEXEC.BAT должна быть внесена следующая команда:

```
c:\dos\mscdex /D:mscd000 /E /L:g
```

В этой команде разрешается использование подключенного драйвера CDROMDRV.SYS. Ключ /E предписывает MSCDEX использовать для размещения секторных буферов расширенную память, если она доступна, а ключ /L:G назначает для CD-ROM диска букву G для его использования в дальнейшем, как логического диска G:

При наличии в вашей системе двух и более дисков CD-ROM в файлы CONFIG.SYS и AUTOEXEC.BAT должны быть включены команды и директивы, как в приводимом ниже примере:

- файл CONFIG.SYS:

```
device = c:\aspic\aspicd.sys /D:mscd000
device = c:\tsl\tslcdr.sys /D:mscd001
```

Каждая из приведенных выше директив подключает свой CD-ROM диск;

- файл AUTOEXEC.BAT. В этот файл должна быть добавлена следующая команда:

```
c:\dos\mscdex /D:mscd000 /D:mscd001 /L:k
```

Эта команда разрешает использование драйверов обоих CD-ROM дисков с признаками MSCD000 и MSCD001. Ключ /L:K предписывает DOS присвоить первому CD-ROM диску обозначение K:, а второму CD-ROM диску будет присвоено обозначение L:.

Примечания:

Число букв, обозначающих логические диски компьютера, ограничивает число CD-ROM дисков, которые может иметь ваша машина. Число логических дисков определяется директивой LASTDRIVE в файле CONFIG.SYS. Во время загрузки MSCDEX буквы, доступные для обозначения логических дисков, не должны использоваться сетевыми дисками или "сжатыми" дисками программы Double Space.

(!) MSD

Получение подробной технической информации о вашем компьютере. Команда внешняя. Файл MSD.EXE. Данная утилита впервые введена в состав DOS, как средство, позволяющее пользователю оперативно

проверить конфигурацию компьютера и просмотреть размещение драйверов устройств и программ в памяти. После запуска программы на экране пользователь видит яркое окно, сообщающее что идет тестирование узлов компьютера, номер версии MSD и дополнительную информацию о работе утилиты. Далее по окончании тестирования экран примет вид, который изображен на рисунке 8.10.

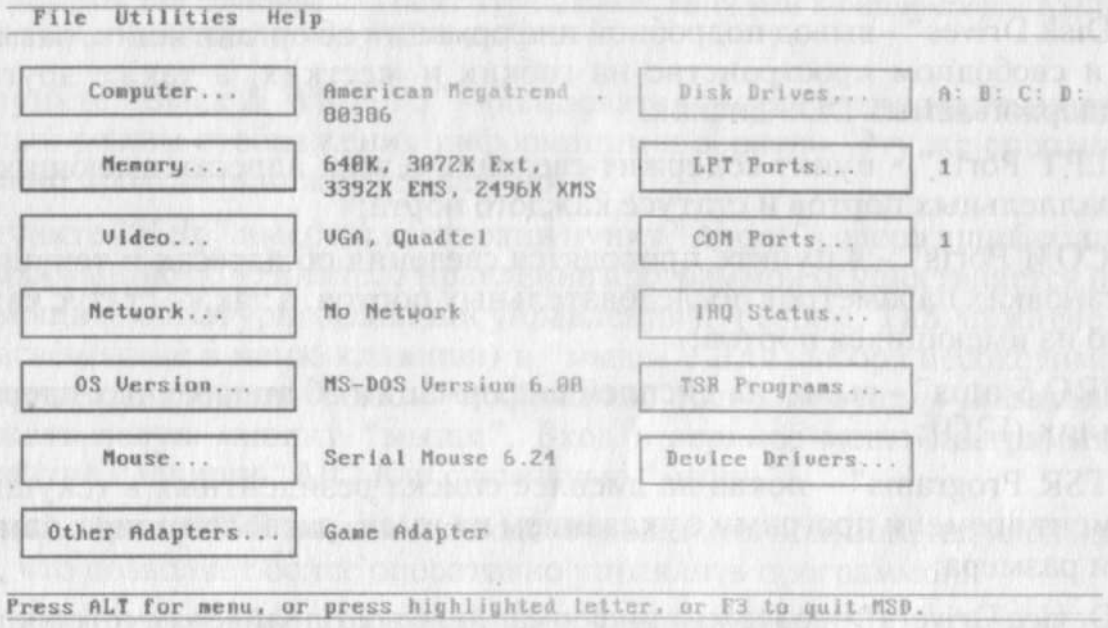


Рис. 8.10

На экране отображаются следующие пункты меню программы:

- "Computer". В этом пункте содержится информация о производителе компьютера, типе процессора и типе системной шины; производителе ROM BIOS, ее версии и дате; типе клавиатуры; конфигурации контроллера прямого доступа в память (DMA - "Data Memory Access" и типе математического сопроцессора;

- "Memory". В этом пункте выводится карта верхней области памяти, в диапазоне от 640K до 1024K;

- "Video" - вывод информации о производителе видеоадаптера, его модели и типе; версии видео BIOS и ее дате, а также о текущем видеорежиме;

- "Network" - пункт содержит сведения о работе вашего компьютера в ЛВС, если компьютер включен в нее;

- "Operating System" - в пункте приводятся сведения о версии операционной системы, расположении ОС MS-DOS в оперативной памяти, системных дисках компьютера, текущих переменных окружения и пути, используемого для запуска утилиты MSD;

- "Mouse". Пункт содержит сведения о версии драйвера манипулятора типа "мышь", типе "мыши", номере используемого для работы "мыши" аппаратного прерывания. Здесь же содержится и другая дополнительная информация по работе с "мышью";

- "Other Adapters". Вывод на дисплей параметров установленного игрового адаптера (если он имеется), а также игровых устройств или джойстиков;

- "Disk Drives" - вывод подробной информации об организации, размере и свободном пространстве на гибких и жестких, а также других поддерживаемых DOS дисках.

- "LPT Ports" - пункт содержит сведения о всех адресах имеющихся параллельных портов и статусе каждого порта;

- "COM Ports" - в пункте приводятся сведения об адресах и текущих установках параметров последовательных портов, а также статус каждого из имеющихся портов;

- "IRQ Status" - вывод на дисплей информации об аппаратных прерываниях (IRQ);

- "TSR Programs" - показ на дисплее списка резидентных в текущий момент времени программ с указанием их имен, расположения в памяти и размера;

- "Device Drivers" - содержит информацию о всех драйверах устройств, установленных в системе на момент запуска Microsoft Diagnostic;

Главное окно MSD содержит также и строку меню вверху экрана. Это меню содержит пункты:

- "File" - работа системными файлами MS-DOS и Microsoft Windows;

- "Utility" - проверка отдельных устройств машины, управление выдачей отчетов о работе MSD, режимом отображения на дисплее;

- "Help" - получение общей информации о программе.

В пункте верхнего меню "File" доступны такие пункты:

- "Find File..." - поиск указанного пользователем. При выборе этого пункта пользователь попадает в новое подменю, где можно выбрать имя файла для поиска, стартовый каталог поиска, задать режим работы с включением в поиск всех подкаталогов - "Include Sub-Dirs", и поиск на всех дисках компьютера - "Search all drives";
- "Print Report..." - в этом пункте осуществляется управление процессом составления отчета о работе MSD. Возможен выбор печати, как по всем параметрам конфигурации компьютера, так и по отдельным темам. Пользователь имеет возможность выбирать для печати различные порты. Печать можно осуществлять также и в специальный файл, которому по умолчанию присвоено имя REPORT.MSD;
  Далее в подменю "File" перечислен целый ряд системных файлов MS-DOS, Windows и других программ, выбрав которую можно просмотреть ее содержание.
- "Exit" - выход из программы MSD.

Пункт "Utility" содержит следующие пункты:

- "Memory Block Display..." - вывод на дисплей карты памяти, с указанием имен всех загруженных на момент запуска Microsoft Diagnostic программ, их сегментных адресов и длины;
- "Memory Browses..." - поиск указанной строки текста в области системного ПЗУ - ROM BIOS и области ROM BIOS видеоадаптера;
- "Insert Command..." - вставка команд в файлы CONFIG.SYS и AUTOEXEC.BAT;
- "Test Printer..." - тестирование принтера. В этом пункте пользователю необходимо выбрать тип принтера - "Printer Type", тип теста (7 или 8-битовые коды ASCII) - пункт "Test Type" и порт - "Printer Port".

В пункте "Black & White F5" - пользователь может установить черно-белый режим отображения информации и обратно. Эту же операцию можно проделать нажатие клавиши F5.

В пункте "Help" имеется всего один пункт "About" содержащий общую информацию об утилите. Управление программой осуществляется при помощи клавиатуры (клавиши управления курсором, Tab, нажатие на подсвеченные в меню клавиши) и "мыши". Для выбора необходимого пункта при помощи "мыши" необходимо подвести е нему указатель и нажать левую кнопку "мыши". Вход в верхнее меню программы - нажатие клавиши "Alt" или с помощью "мыши".

Внизу экрана MSD постоянно подсвечивает для пользователя подсказку, что позволяет более оперативно управлять программой.

При использовании традиционного управления MSD из командной строки, формат команды имеет вид:

msd /I /F*диск:\маршрут\имя_файла* /P*диск:\маршрут\имя_файла* /S*диск:\маршрут\имя_файла*

Приведенный выше формат команды используется для создания MSD отчета. Для того, чтобы MSD проверила конфигурацию машины с использованием своего интерфейса, необходимо задавать ее в виде:

```
msd /B /I
```

Параметры:

диск:\маршрут\имя_файла - указание полного идентификатора файла, который будет хранить отчетную информацию.

Ключи

/I	при использовании этого ключа MSD не будет проверять оборудование компьютера. Указывайте этот ключ, когда у вас имеются проблемы с запуском программы MSD при ее старте или если она не запускается вообще;

/F*диск:* *\маршрут* *\имя_файла*	запись в указанный отчетный файл имени пользователя, названия компании, адреса, страны, примечаний и телефонных номеров;
/P*диск:* *\маршрут* *\имя_файла*	запись в указанный отчетный файл полной отчетной информации без вывода справочной информации;
/S*диск:* *\маршрут* *\имя_файла*	запись всей выходной информации MSD сразу в указанный файл, а не на экран. По умолчанию MSD выводит всю свою информацию на экран;
/B	запуск Microsoft Diagnostic в черно-белом режиме отображения информации. Этот ключ необходимо использовать в тех случаях, когда имеются проблемы вывода информации на цветной монитор.

Примеры использования:

Допустим, необходимо, чтобы в процессе работы программы MSD был создан файл с именем PC.TXT, включающий полный отчет о технических средствах компьютера. Для решения этой задачи введите команду:

```
msd /P pc.txt
```

Если же вам необходимо, чтобы программа MSD создала детальный отчет о конфигурации вашего компьютера, с включением в него вашего имени, компании, адреса, примечаний, телефонного номера, необходимо ввести следующую команду:

```
msd /F pc.txt
```

NLSFUNC

Поддержка MS-DOS национальных форматов. Программа NLSFUNC загружается в память для того, чтобы обеспечить в дальнейшем переключение кодовых страниц командой CHCP. Команда внешняя. Файл NLSFUNC.EXE. Команда NLSFUNC может быть задана в файле CONFIG.SYS для поддержки национальных форматов или загружена в память из командной строки DOS. Команду нельзя использовать при работе в Windows.

Формат команды:

nlsfunc *диск:\маршрут\имя_инф_файла*

При использовании команды NLSFUNC в файле CONFIG.SYS необходимо использовать следующий формат:

install=*диск1:\маршрут1*\nlsfunc.exe *диск1:*
\маршрут\имя_инф_файла

Параметры:

диск:\маршрут\имя_инф_файла - указание полного идентификатора файла, содержащего информацию о национальных кодовых страницах и форматах. При задании имени информационного файла в файле CONFIG.SYS необходимо указывать его полный идентификатор, включающей букву диска и каталог(ги);

диск1:\маршрут1 - указание местоположения файла NLSFUNC.EXE.

Примеры использования:

Для использования информации о национальных формата по умолчанию из файла CONFIG.SYS, необходимо ввести команду:

```
nlsfunc
```

Допустим вы имеете файл INFORM.SYS, содержащий информацию о специфических национальных форматах и кодовых страницах. Для использования информации из него при переключении кодовых страниц можно либо подключить его к ОС командой NLSFUNC в файле CONFIG.SYS, либо ввести из командной строки DOS следующую команду:

```
nlsfunc inform.sys
```

Примечания:

- Место расположения файла COUNTRY.SYS обычно определяет директива COUNTRY файла CONFIG.SYS. Если команда COUNTRY в файле CONFIG.SYS не используется, или файл COUNTRY.SYS не найден в указанном каталоге, то утилита NLSFUNC ищет его в корневом каталоге системного диска.
- В том случае, если файл COUNTRY.SYS не будет найден MS-DOS в каталоге, указанном в команде NLSFUNC, сообщение об ошибке выведено не будет. Однако, когда вы попытаетесь переключить кодовые таблицы командой CHCP будет выведено сообщение об ошибке.

Подробности применения команды CHCP приведены в ее описании. Информация о подготовке кодовых страниц перед их загрузкой в память содержится в описании команды MODE CP PREP.

PATH

Задание маршрутов поиска выполняемых файлов MS-DOS (имеющих расширения .COM, .EXE, .BAT) в каталогах указанных дисков. При нахождении выполняемого файла в каталоге, указанном в команде PATH, он может быть запущен на выполнение из любого каталога любого диска. По умолчанию поиск указанного выполняемого файл выполняется в текущем каталоге. Команда внутренняя.

Формат команды:

path *диск:\маршрут;...*

Для вывода на дисплее текущего маршрута поиска команды PATH, используйте следующий вариант команды:

path

Для очистки всех маршрутов поиска PATH и установления маршрута поиска по умолчанию (текущий каталог), команду необходимо задать в виде:

path ;

Параметры:

диск:\маршрут - указание дисков, каталогов и подкаталогов для организации процесса поиска выполняемых файлов. Каталоги и диски поиска файлов должны разделяться между собой символом ";".

Примеры использования:

Для указания маршрута поиска файлов в трех каталогах C:\DOS, C:\USER и D:\BIN необходимо ввести команду PATH в следующем виде:

```
path c:\dos;c:\user;d:\bin
```

Примечания:

- При организации поиска файлов команда PATH сначала осуществляет поиск файлов в текущем каталоге, и только потом в каталогах, перечисленных в команде PATH.
- Длина строки маршрутов поиска, перечисленных в команде PATH, не может превышать 127 символов. Для включения как можно большего числа каталогов в маршруты поиска необходимо использовать короткие имена каталогов, команду SUBST для переназначения каталогов в логические диски (с дальнейшим использованием букв логических дисков в команде PATH), или использовать команду APPEND с ключом /X:ON.

- При организации поиска файлов MS-DOS сначала осуществляет поиск указанного файла с расширением .COM, затем .EXE и только потом файла с расширением .BAT.

  Допустим, если в каком-то каталоге имеется два файла с одинаковым именем, например PRIMER, но с разными расширениями .COM и .BAT, и вам необходимо запустить на выполнение файл с именем PRIMER.BAT. Если вы просто введете имя файла PRIMER, то загружен в память будет файл PRIMER.COM, а не файл PRIMER.BAT. Чтобы запустить файл с именем PRIMER.BAT необходимо либо располагать его в других каталогах, либо вводить его полное имя.
- Если команда PATH включает в себя несколько маршрутов поиска файлов, то при нахождении в разных каталогах файлов с одним и тем же именем, на выполнение будет запущен тот, который будет найден раньше.
- Команду PATH очень удобно использовать в файле AUTOEXEC.BAT для настройки маршрутов поиска при загрузке компьютера.

(!) POWER

Управление питанием устройств компьютера, вывод на дисплей отчета об управлении питанием устройств компьютера и установка уровней сохранения энергии источника питания компьютера. Утилита разработана и применяется в основном для портативных компьютеров класса "Laptop", "Notebook" и "Palmtop". Команда внешняя. Файл POWER.EXE.

Формат команды:

power ADV:MAX|REG|MIN|STD|OFF

Для вывода на дисплей текущего статуса утилиты POWER, используйте следующий формат команды:

power

Параметры:

ADV:MAX|REG|MIN - сохранение запаса мощности источника питания во то время, когда прикладные программы и устройства простаивают.

Параметр MAX используется для максимального сохранения емкости источника питания.

Параметр REG действует по умолчанию, он позволяет сбалансировать потребление энергии между устройствами при простое и работе прикладных программ.

Параметр MIN сводит к минимуму сохранение энергии при простое прикладных программ. Используйте параметр MIN в том случае, если использование параметров MAX и REG не приносит желаемого эффекта.

STD - этот ключ используется в том случае, если оборудование компьютера соответствует спецификации APM (Advanced Power Managment - расширенное управление питанием). При использование этого ключа DOS будет использовать для сохранения энергии источника питания аппаратные средства компьютера. В том случае, если компьютер не поддерживает спецификацию APM, параметр STD выключает управление питанием.

OFF - отключение управления питанием компьютера.

Примечания:

Перед использованием команды POWER вы должны установить в файле CONFIG.SYS драйвер POWER.EXE директивой DEVICE.

Для получения дополнительной информации о драйвере POWER.EXE необходимо просмотреть описание этого драйвера в главе "Загружаемые драйверы устройств".

PRINT

Печать текстовых файлов в фоновом режиме во время выполнения DOS других команд. Команда позволяет печатать тексты на одном из подсоединенных к компьютеру принтеров, через параллельный или последовательный порт. Команда внешняя. Файл PRINT.EXE.

Формат команды:

print /D:*устройство* /B:*размер* /U:*такт1* /M:*такт2* /S:*такт3* /Q:*число_файлов* /T *диск:\маршрут\имя_файла...* /C /P

Для установки параметров утилиты PRINT по умолчанию или вывода на дисплей очереди печати, используйте следующий формат команды:

print

Параметр:

диск:\маршрут\имя_файла... - указание полного идентификатора файла или нескольких файлов, которые необходимо вывести на печать. В одной строке можно указывать до 10 файлов одновременно.

Ключи

/D:*устройство* указание имени устройства. Допустимыми значениями для параллельных принтеров являются следующие LPT1, LPT2, LPT3. Для последовательных портов допустимыми значениями являются следующие COM1, COM2, COM3, COM4. По умолчанию принимается значение LPT1 или синоним PRN. Ключ /D:*устройство* должен предшествовать указанию имен файлов, подлежащих выводу на печать;

/B:*размер* установка в байтах размера внутреннего буфера, в котором хранятся данные перед их выводом на принтер. Минимальное значение для этого параметра равно 512, оно же установлено и по умолчанию. Максимальная величина буфера может быть равной 16384. Увеличение размера буфера увеличивает скорость печати под управлением PRINT, но уменьшает объем памяти, доступной другим программам.

/U:*такт1* задание максимального числа тактов системного таймера, в течение которого система будет ожидать появления сигнала готовности от принтера. (1 такт таймера равен 55 микросекунд). Если принтер в течение указанного времени не выставит сигнала готовности, печать информации не будет продолжена. Значение параметра может изменяться в пределах от 1 до 255. По умолчанию параметр *такт1* установлен в 1;

/M:*такт2* указание максимального числа тактов системного таймера, в течение которого ОС будет отдавать управление программе PRINT для вывода символов на печать; Значения параметра *такт2* меняется в пределах от 1 до 255. По умолчанию параметр *такт2* равен 2. Если символ печатается слишком медленно, MS-DOS выводит сообщение об ошибке;

/S:*такт3* число тактов процессорного времени, пропускаемых программой PRINT перед каждым обращением к центральному процессору. Значение параметра *такт3* по умолчанию установлено равным 8 и меняется от 1 до 255. Возрастание значения этого параметра увеличивает скорость работы других программ при их совместной работе с утилитой PRINT;

/Q:*число_файлов* указание максимального числа файлов, одновременно включенных в очередь печати. Значение па-

раметра *число_файлов* должно находится в пределах от 4 до 32. По умолчанию это значение равно 10;

/T — удаление текущего файла из очереди печати и очищение очереди к принтеру;

/C — удаление всех файлов из очереди печати. Ключи /C и /P могут совместно использоваться в одной командной строке. Когда вы указываете DOS, что необходимо удалить из очереди печати файл с ключом /C, DOS удаляет из очереди файл с этим ключом и все файлы после него, до тех пор пока не встретится файл с ключом /P;

/P — добавление файлов в очередь печати. Ключ действует на файл, имеющий ключ /P, и на все файлы, которые следуют за этим ключом.

Примеры использования:

При использовании команды PRINT в виде:

```
print
```

MS-DOS выводит на дисплей имя каждого файла, который отпечатан, имена файлов, находящихся в очереди и сообщения об ошибке, если такие ошибки имеются. Для исключения файла BOOKS.TXT из очереди печати необходимо ввести команду:

```
print c:\text\books.txt /C
```

Следующий пример иллюстрирует, как исключить из очереди файл BOOKS.TXT и добавить в очередь печати файл README.DOC:

```
print books.txt /C readme.doc /P
```

Следующие примеры показывает, как необходимо настраивать параметры команды PRINT при запуске из командной строки MS-DOS.

Для направления печати на второй параллельный порт LPT2 введите команду:

```
print /D:lpt2
```

Для того чтобы команда PRINT ожидала 80 тактов сигнала готовности от принтера и пропускала 35 тактов при очередном обращении к процессору во время печати файлов, необходимо ввести команду:

```
print /U:80 /S:35
```

Если вам необходимо, чтобы ОС отдавала на 6 тактов управление программе PRINT для печати символов, команда PRINT должна быть введена в виде:

```
print /M:6
```

Для изменения количества файлов, одновременно находящихся в очереди печати с 10 (по умолчанию) на 32, введите команду:

```
print /Q:32
```

Примечания:

- Каждый элемент очереди печати может включать максимум 64 символа (букву диска, каталог и подкаталоги)
- Ключи /D, /B, /U, /M, /S и /Q могут быть использованы при первом запуске программы PRINT при запуске из командной строки MS-DOS. Для изменения одного из этих ключей необходимо перезапустить компьютер и ввести команду PRINT с измененными ключами.
- Файлы выводятся на печать в том порядке, в котором они указаны в команде PRINT.
- В процессе использования команды PRINT нельзя использовать принтер для печати других документов.

PROMPT

Изменение вида системного запроса DOS. Команда внутренняя. Используя эту команду, вы можете изменить стандартный запрос DOS в виде символа ">" на любой текст, а также вывести имя текущего каталога, времени и текущей даты, а также версии MS-DOS.

Формат команды:

prompt *текст*

Параметр:

текст - задание текста или строки символов, которые необходимо включить в запрос MS-DOS. Допускается использование в параметре *текст* специальных символов, обозначения которых и функции, решаемые ими, приведены ниже:

$Q - вывод на дисплей символа "=";

$$ - вывод на дисплей символа "$";

$T - вывод на дисплей текущего времени;

$D - вывод на дисплей текущей даты;

$P - вывод на экран буквы текущего диска и имени текущего каталога;

$V - индикация на экране версии MS-DOS;

$N - индикация буквы текущего диска;

$G - индикация символа ">";

$L - индикация символа "<";

$B - индикация символа "|";

$_ - переход в начало следующей строки;

$E - индикация символа Esc (код ASCII 27);

$H - стирание предыдущего символа, выведенного в приглашении DOS.

Примеры использования:

Для того, чтобы в системном запросе присутствовало обозначение текущего диска, имя текущего каталога и символ ">", команду PROMPT следует задать в следующем виде:

```
prompt $p$g
```

Для включения в системный запрос строки "Hello friend !", также отображения системной даты и времени, необходимо ввести команду:

```
prompt hello friend ! time:$t date:$d
```

Если в вашем файле CONFIG.SYS к ОС подключен драйвер ANSI.SYS, возможности управления видом системного запроса расширяются. В следующем примере показывается, как с помощью команд ANSI.SYS выделить подсказку DOS ярким, а всю остальную информацию нормальным цветом.

```
prompt $e[1m$n:$e[m
```

Символы $E начинают Esc-последовательность команд драйвера ANSI.SYS.

Примечания:

- Для сброса всех установок команды PROMPT необходимо ввести ее без параметра. В этом случае все предыдущие установки PROMPT будут сброшены и запрос DOS примет вид по умолчанию - символ ">".
- При работе в среде Microsoft Windows 3.1 пользователь может изменить вид системного запроса, изменив окружение переменной WINPMT. Для примера, если вам необходимо, чтобы системный запрос в сеансе MS-DOS имел следующий вид:

  ```
  Please type "exit" to return to Windows 3.1
  ```

  Для решения этой задачи в файл AUTOEXEC.BAT необходимо включить следующую строку:

```
set winpmt = Please type "exit" to return to Windows
3.1.$_$p$t$g
```

- Для дополнительной информации об отображении системного времени и даты просмотрите описание команд DATE и TIME.
- Список команд драйвера ANSI.SYS приведен в Приложении 3.

RENAME (REN)

Переименование одного или нескольких файлов, указанных пользователем. Команда внутренняя. Команда позволяет переименовывать любые файлы, указанные пользователем, на всех доступных дисках компьютера. Команда не позволяет перемещать файлы и переименовывать каталоги. Для этих целей в составе MS-DOS имеется команда MOVE.

Форматы команды:

rename *диск:\маршрут\имя_файла1 имя_файла2*
ren *диск:\маршрут\имя_файла1 имя_файла2*

Параметры:

диск:\маршрут\имя_файла1 - указание имени файла или файлов, подлежащих переименованию;

имя_файла2 - указание нового имени файла или файлов, которое файл (файлы) получат после переименования.

Примеры использования:

Если вам необходимо, чтобы все файлы с любыми именами, но с расширением .TXT были переименованы в файлы с теми же именами, но с расширением .DOC, введите команду:

```
ren *.txt *.doc
```

Для переименования файла TEST.LST с диска D: на файл с именем TEST.RPT, введите команду:

```
ren d:test.lst test.rpt
```

Файл с новым именем TEST.RPT будет размещен на том же диске D:

Примечания:

- При отсутствии в команде REN параметра имя_файла2 эта команда выполняться не будет.
- В том случае, когда имя файла в параметре *имя_файла2* совпадает с именем файла в параметре *имя_файла1*, появится сообщение:

```
Duplicate file name or file not found
```

REPLACE

Выборочная замена файлов в одном каталоге на файлы из другого каталога с теми же именами. Команда может использоваться также для добавления файлов из одного каталога в другой. Команда внешняя. файл REPLACE.EXE.

Формат команды:

replace *диск1:\маршрут1\имя_файла1 диск2:\маршрут2* /A /P /R /W
replace *диск1:\маршрут1\имя_файла1 диск2:\маршрут2* /P /R /S /W /U

Параметры:

диск1:\маршрут1\имя_файла1 - указание местоположения файла или группы файлов, предназначенных для замены файлов другого каталога или добавления в каталог;

диск2:\маршрут2 - указание диска и каталога, файлы которого будут заменены или каталога, в который будут добавлены новые файлы. Если этот параметр не указан, то команда REPLACE использует для замены или добавления файлов текущий каталог;

Ключи

/A	добавление в указанный каталог только тех файлов, которые отсутствуют в указанном параметром *диск:\маршрут2* каталоге. Этот ключ не допускается использовать совместно с ключами /S и /U;
/P	этот ключ предписывает DOS спрашивать разрешения у пользователя на замену или добавление каждого файла;
/R	при использовании этого ключа заменяются и файлы, имеющие атрибут "только для чтения". Если этот ключ не используется, то выполнение команды REPLACE приведет к появлению сообщения об ошибке и прекращению замены файлов;
/S	при использовании этого ключа будут заменены все файлы в указанном каталоге, а также в подчиненных каталогах. Вы не можете использовать совместно с этим ключом ключи /S и /A;
/W	организация паузы перед установкой исходной дискеты. Если ключ не используется, команда REPLACE начинает замену или добавление файлов сразу после нажатия клавиши Enter;
/U	замена только тех файлов в каталоге, указанном в параметре *маршрут2*, которые не существуют в исходном каталоге для

замены (параметр *маршрут1*). Ключ /U не допускается совместно использовать с ключом /A;

Примеры использования:

Допустим, на диске C: в различных каталогах расположены различные варианты файла NOTEBOOK.APP, который содержит тест прикладной программы. Для замены всех вариантов этого файла на последнюю его версию, расположенную на диске A:, необходимо ввести команду:

```
replace a:\notebook.app c:\ /S
```

Команда:

```
replace a:\*.* b: /S /R
```

заменяет файлы, расположенные на диске B: на файлы диска A:, расположенные во всех каталогах подчиненных корневому, с теми же именами и расширениями. Замена производится и для файлов, имеющих атрибут "только для чтения". Для добавления с диска A: файлов в дополнение к существующим в корневом каталоге диска C: необходимо ввести команду:

```
replace a:*.* c:\ /A
```

Примечания:

- После добавления или замены командой REPLACE файлов DOS выведет на дисплей отчетное сообщение о работе утилиты REPLACE в следующем формате:

  XXX files added
  XXX files replaced
  (XXX файлов добавлено)
  (XXX файлов заменено)

  либо

  XX file added
  XX file replaced
  (XX файлов добавлено)
  (XX файлов заменено)

- При замене файлов с использованием гибких дисков желательно использовать ключ /W для организации нормальной смены дискет.
- Команда REPLACE не может быть использована для работы с "системными" и "скрытыми" файлами, такими например, как IO.SYS и MSDOS.SYS.

- По окончании своей работы REPLACE возвращает DOS следующие коды завершения:

0	успешная замена или добавление утилитой REPLACE файлов;
1	компьютер работает под управлением MS-DOS, версия которой несовместима с версией команды REPLACE;
2	REPLACE не нашла указанных для замены файлов;
3	утилита REPLACE не нашла путей доступа к маршруту, указанному в параметре *маршрут1* или в параметре *маршрут2*;
5	пользователь не может получить доступа к файлам, указанным для замены;
8	недостаточно оперативной памяти системы для выполнения команды;
11	использование неправильного формата команды.

Коды завершения команды REPLACE могут использоваться в дальнейшем в командных файлах или в прикладных программах.

(!) RESTORE

Восстановление файлов резервного копирования, созданных с использованием команды BACKUP MS-DOS версий от 2.0 до 5.0.

Для восстановления файлов резервного копирования, созданных утилитой MS-DOS 6.0 MSBACKUP, необходимо пользоваться той же утилитой. Команда внешняя. Файл RESTORE.EXE.

Формат команды:

restore *диск1: диск2:\маршрут\имя_файла* /S /P /B:*дата* /A:*дата* /E:*время* /L:*время* /M /N /D

Параметры:

диск1: - указание буквы диска, содержащего файл резервного копирования.

диск2: - указание буквы диска, на котором необходимо восстановить файлы, скопированные программой BACKUP.

маршрут - указание каталога, в который будет производиться запись восстановленных файлов. Имя каталога должно совпадать с тем, которое было присвоено каталогу, содержащему восстанавливаемые файлы до резервного копирования.

имя_файла - указание имени или имен файлов, подлежащих восстановлению.

Ключи

/S	восстановление файлов, включая файлы во всех подкаталогах;

/P — перед восстановлением файла, имеющего атрибут "только для чтения" или измененного со времени последнего резервного копирования, имеющего атрибут "архивный", DOS будет запрашивать разрешение перед восстановлением каждого файла;

/B:*дата* — восстановление только файлов, созданных или модифицированных в указанный день и до него. Формат даты в ключе должен соответствовать установленному директивой COUNTRY в файле CONFIG.SYS. Для получения информации о форматах даты следует обратиться к описании команды DATE;

/A:*дата* — восстановление только файлов, созданных в указанный день или после него. Формат даты должен соответствовать установленному директивой COUNTRY файла CONFIG.SYS;

/E:*время* — восстанавливаются файлы, созданные в указанное время или до него. Формат времени должен соответствовать установленному в директиве COUNTRY файла CONFIG.SYS. Для получения дополнительной информации о команде RESTORE необходимо изучить описание команды TIME;

/L:*время* — восстановлению подвергаются только файлы, созданные в указанный момент времени или после него. Формат времени должен соответствовать установленному директивой COUNTRY в файле CONFIG.SYS;

/M — восстановление файлов, модифицированных или удаленных с момента последнего резервного копирования;

/N — восстанавливаются файлы, которые не существуют на указанном диске;

/D — вывод на дисплей списка файлов на диске с резервными файлами, совпадающими с указанными. Даже если файлы не будут восстановлены, при использовании параметра *диск2:* следует использовать ключ /D.

Примеры использования:

Для восстановления файла с именем IMPORT.DAT с файла резервного копирования на диске A: в каталог DATA диска C: необходимо ввести команду:

```
restore a: c:\data\import.dat
```

После ввода команды MS-DOS попросит вставить диск резервного копирования в дисковод A:. После того, как диск с резервной копией вставлен в диск A:, необходимо нажать клавишу Enter. Для восстановления всех файлов в каталог \SYSTEM\RPT диска C: с диска A:, необходимо ввести команду:

```
restore a: c:\system\rpt\*.*
```

Для восстановления полной древовидной структуры каталогов диска с диска (дисков) резервного копирования, например с диска A:, необходимо ввести команду:

```
restore a: c:\*.* /S
```

Ключ /S, используемый совместно с групповым шаблоном *.*, указывает команде RESTORE на необходимость восстановления всех файлов во всех каталогах, подчиненных корневому каталогу диска C:.

Примечания:

- Для проверки соответствия восстановленных файлов оригиналу необходимо использовать команды DIR или TYPE;
- Команда RESTORE не может использоваться для восстановления "системных" файлов DOS - IO.SYS и MSDOS.SYS. RESTORE не работает с дисками, переназначенными командами ASSIGN и JOIN предыдущих версий DOS.
- По окончании своей работы RESTORE возвращает DOS следующие коды завершения:

0	RESTORE успешно завершил восстановление файла или файлов;
1	утилита RESTORE не смогла найти файлы для восстановления;
3	пользователь нажал клавишу Ctrl-C для прекращения процесса восстановления файлов;
4	утилита RESTORE остановила работу по причине обнаружения ошибки во время ее работы.

  Коды завершения команды RESTORE используются обычно в параметре ERRORLEVEL командных файлов или в прикладных программах.
- Файлы, которые были скопированы командой BACKUP на одном диске, могут быть восстановлены также и на другом диске. Для примера, файлы резервного копирования с диска C: каталога \FILES могут быть восстановлены на диске D: в каталоге с тем же именем.

RMDIR (RD)

Удаление каталога. Команда внутренняя. Перед удалением каталога вы должны очистить его и все подчиненные каталоги от файлов.

Форматы команды:

rmdir *диск:\маршрут*
rd *диск:\маршрут*

Параметр:

диск:\маршрут - указание местоположения и имени удаляемого каталога.

Примеры использования:

Для удаления каталога с именем \SYSTEM\MYFILE необходимо сначала ввести команду:

```
dir \system\myfile /A
```

для просмотра его содержимого.

На дисплей MS-DOS выведет только символы "." и "..", говорящие, что каталог \SYSTEM\MYFILE пуст. После этого можно ввести команду:

```
rmdir \system\myfile
```

Поставленная раньше задача может быть решена вводом команды RMDIR в другом виде:

```
rd \system\myfile
```

Примечания:

- Нельзя удалить каталог, включающий "скрытые" и "системные" файлы без принятия специальных мер. При попытке удаления каталога, включающего "системные" и "скрытые" файлы, на дисплее появится сообщение:

  ```
  Invalid path, not directory,
  or directory not empty
  (Неправильный путь, нет каталога
  или каталог не пуст)
  ```

- В этом случае необходимо использовать команду DIR с ключами, позволяющими вывести список "системных" и "скрытых" файлов, а также команду ATTRIB для изменения атрибутов файлов. После изменения атрибутов "системных" и "скрытых" файлов они могут быть удалены командой DEL (ERASE).

- При использовании в параметре *маршрут* символа "\", DOS начинает поиск указанного каталога от корневого каталога. Если же этот символ в параметре *маршрут* не указан, то поиск осуществляется от текущего каталога.
- Команда RD не позволяет удалить пользователю текущий каталог. При попытке удаления текущего каталога на дисплей будет выведено сообщение:

  Attempt to remove current directory - drive:path
  (Попытка удаления текущего каталога - диск:\маршрут)

- Команда RMDIR не позволяет удалить каталоги, переназначенные командой SUBST. Командой RMDIR нельзя также удалить корневой каталог.
- Для получения дополнительной информации об удалении каталогов, включая все файлы в них, подчиненные каталоги и файлы подчиненных каталогов, необходимо обратиться к описанию команды DELTREE.

SET

Установка, изменение или вывод на дисплей переменных окружения MS-DOS. Команда внутренняя. Переменные окружения очень часто используются в командных файлах и программах для осуществления управления MS-DOS и контроля ее работы. Подробные сведения о системном окружении представлены в разделе "Системное окружение DOS" главы "Состав MS-DOS и ее основные модули". Обычно команда SET применяется в составе файлов CONFIG.SYS и AUTOEXEC.BAT, создающих индивидуальную для каждого пользователя рабочую среду.

Формат команды:

set *переменная* = *строка*

Для вывода на дисплей текущего статуса переменных окружения, команда задается в формате:

set

Параметры:

переменная - указание имени переменной DOS для установки или изменения;

строка - задание строки символов, связанных с указанной переменной определенным смыслом. Параметр *строка* может содержать, например, маршрут доступа к какому-то файлу ОС или прикладной программы.

Примеры использования:

Для установки переменной окружения TEMP и задания для нее смысловой строки в виде маршрута доступа к файлам C:\WINDOWS\TEMP введите команду:

```
set temp = c:\windows\temp
```

Очень часто команда SET используется для указания пути к командному процессору:

```
set comspec = c:\dos\command.com
```

Другим распространенным вариантом использования команды SET является ее применение в командных файлах, как в приведенном ниже примере:

```
@echo off
rem Файл ADDDIR.BAT добавляет указанный каталог
rem в окружение системной переменной PATH
set path=%1;
set
```

Примечания:

- При вводе команды SET без параметров MS-DOS выводит на дисплей сообщение, включающее список системных переменных, устанавливаемых по умолчанию. В этот список всегда входят переменные COMSPEC и PATH. Переменные PROMPT и DIRCMD также обычно включаются в состав переменных системного окружения. Дополнительную информацию о переменной DIRCMD можно найти в описании команды DIR.
- При добавлении переменной в состав окружения DOS, если эта переменная уже существует, происходит замена старых значений переменной на новые.
- Если добавляемая в состав окружения переменная имеет символ "=" без указания параметра "строка", то все значения существующей переменной очищаются.
- При задании значений системной переменной в пакетном файле командой SET эту переменную можно в дальнейшем использовать в командной строке DOS, заключая в символы "%". Для примера, если в пакетном файле установлена переменная STOP, то текстовая строка, с ней ассоциируемая, может быть вызвана из командной строки набором символов:

  ```
  %stop%
  ```

- В случае нехватки места для размещения очередной переменной своего окружения DOS выведет на дисплей следующее сообщение:

  Out of environment space
  (Конец области окружения)

- В этом случае необходимо увеличить размер системного окружения, и попробовать вновь задать командой SET необходимую переменную окружения. Как увеличить размер системного окружения рассказывается в описании команды COMMAND.

SETVER

Объявление версии операционной системы. Команда используется для обеспечения работы программ, требующих наличия предыдущих версий DOS.

Команда также используется для изменения версии ОС и вывода на дисплей списка команд с указанием номеров версий DOS, ими используемых.

Команда внешняя. Файл SETVER.EXE.

Формат команды:

setver *диск:\маршрут\имя_файла n.nn*
setver *диск:\маршрут\имя_файла* /DELETE /QUIET

Для вывода на дисплей номера версии DOS, которую использует команда, используйте утилиту SETVER в виде:

setver *диск:\маршрут*

Выходная информация MS-DOS представлена двумя столбцами: слева приведен список имен программных файлов, справа отображается номер версии DOS, соответствующий каждому программному файлу, загруженному в память.

Параметры:

диск:\маршрут - указание местоположения файла, для которого необходимо объявить версию операционной системы.

имя_файла - указание имени программы, обычно имеющей расширение .EXE или .COM, имя которой необходимо добавить в таблицу версий ОС. В этом параметре необходимо указывать имя только одной программы. Использование групповых шаблонов "*" и "?" не допускается.

n.nn - задание номера версии ОС, для примера 3.30, которую необходимо сообщить прикладной программе.

Ключи

/DELETE	удаление таблицы версий MS-DOS для конкретного программного файла. Это ключ можно заменить аббревиатурой /D;
/QUIET	использование этого ключа подавляет вывод сообщений ОС при удалении таблицы версий ОС для конкретного файла. Ключ имеет смысл только при совместном использовании с ключом /DELETE.

Примеры использования:

Допустим, программа MYPROG.EXE нормально функционирует только при наличии MS-DOS версии 3.30. Для того, чтобы запустить эту программу в среде MS-DOS версии 6.00, необходимо перед ее запуском ввести команду:

```
setver myprog.exe 3.30
```

Для удаления имени файла MYPROG.EXE из таблицы версий MS-DOS, необходимо ввести команду:

```
setver myprog.exe /delete
```

Если имеется необходимость вывести таблицу версий для диска C:, введите следующую команду:

```
setver c:
```

Примечания:

- Соответствия прикладных программ версиям DOS хранятся в таблице версий программы SETVER. Эта таблица загружается в память системы только при загрузке компьютера. При внесении изменений в таблицу версий команды SETVER, изменения начнут действовать только после перезагрузки системы.
- Для использования команды SETVER, необходимо подключить файл SETVER.EXE к операционной системе директивой DEVICE файла CONFIG.SYS. По умолчанию, MS-DOS всякий раз настраивает таблицу версий при загрузке операционной системы, что гарантирует нормальное выполнение большинства программ.
- Команда SETVER может использоваться для установки номера версии командного процессора. В том случае, если вы установили командой SETVER номер версии MS-DOS 6.0 для командного процессора, следует воздерживаться от рестарта вашей системы.
- Если с помощью команды SETVER предполагается настраивать драйверы устройств в файле конфигурации системы, то эта строка должна быть первой в CONFIG.SYS.

- При указании в команде SETVER имени программы, ранее внесенной в таблицу версий SETVER, новые данные будут записаны поверх старых.
- По окончании своей работы SETVER возвращает DOS следующие коды завершения:

0	утилита SETVER успешно выполнила поставленные задачи;
1	пользователь указал неправильные ключи в команде;
2	пользователь использовал неправильное имя файла;
3	недостаточно объема оперативной памяти для выполнения команды;
4	пользователь использовал в команде неверный формат для указании версии MS-DOS;
5	SETVER не нашла указанную строку в таблице версий;
6	команда SETVER не смогла найти файла SETVER.EXE;
7	пользователь использовал неверную букву - идентификатор диска;
8	пользователь указал слишком много параметров в командной строке;
9	в командной строке указано недостаточное количество параметров;
10	утилита SETVER обнаружила ошибки во время чтения файла SETVER.EXE;
11	файл SETVER.EXE испорчен;
12	указанный в командной строке SETVER.EXE файл не поддерживает таблицу версий;
13	недостаточно места в таблице версий для размещения очередного имени файла;
14	утилитой SETVER обнаружена ошибка, во время записи информации в файл SETVER.EXE.

Коды завершения SETVER в дальнейшем могут быть использованы для работы с командными файлами и в прикладных программах.

SHARE

Запуск утилиты SHARE, которая контролирует процесс запирания разделяемых файлов и записей на локальных и сетевых дисках компьютера.

Команда поддерживает систему запирания файлов и записей при их одновременном использовании несколькими программами, выполняемыми на одном компьютере.

Команда внешняя. Файл SHARE.EXE.

Формат команды:

share /F:*место* /L:*макс*

При установке утилиты в файле CONFIG.SYS необходимо использовать следующий формат команды:

install = *диск:\маршрут*\share.exe /F:*место* /L:*макс*

Параметр:

диск:\маршрут - указание местоположения файла SHARE.EXE.

Ключи

/F:*место*	указание объема памяти, выделяемого для хранения информации о разделяемых файлах в байтах. По умолчанию этот параметр равен 2048 байт;
/L:*макс*	указание максимального числа одновременно запертых файлов или записей. По умолчанию это число равно 20.

Примеры использования:

Следующий пример показывает, как необходимо использовать директиву INSTALL для загрузки файла SHARE.EXE, в процессе обработки DOS файла CONFIG.SYS. Например, пусть файл SHARE.EXE находится в каталоге DOS диска C:. Команда

```
install = c:\dos\share.exe
```

позволит загрузить команду SHARE с параметрами, заданными по умолчанию.

```
install = c:\dos\share.exe /F:4096 /L:25
```

Эта команда резервирует в оперативной памяти область, объемом в 4096 байт для хранения служебной информации о разделяемых файлах и позволяет одновременно держать запертыми 25 файлов.

Примечания:

- Команда SHARE чаще всего находит применение в компьютерах, использующих сетевые или многозадачные средства запирания разделяемых файлов. Основное назначение команды состоит в запирании доступа к файлам из других программ (от других пользователей), в случае если одна программа или пользователь уже работает с файлом. Такое запирание файлов необходимо, чтобы несколько программ не смогли осуществить одновременную запись в файл информации.
- При определении числа байтов, которые необходимо отвести для записи под один файл, необходимо учитывать количество символов, включенных в полный спецификатор диска. Обычно полный идентификатор файла составляет от 15 до 20 символов.
- Утилита SHARE должна быть использована для работы приложений в среде Microsoft Flash File System ("мерцающей" файловой системы Microsoft).

(!) SMARTDRV

Загрузка или конфигурирование утилиты SMARTDRV, создающей дисковый кэш в дополнительной памяти. Использование дискового кэша позволяет значительно увеличить скорость выполнения дисковых операций. Для загрузки SMARTDRV.EXE можно использовать, как файл CONFIG.SYS, так и файл AUTOEXEC.BAT, а также командную строку DOS. SMARTDRV.EXE может включить также двойную буферизацию для контроллеров жестких дисков, не работающих совместно с драйвером EMM386 или в расширенном режиме 386 процессора для Windows - ("386 Enhanced Mode").

Для включения двойной буферизации необходимо подключить SMARTDRV директивой DEVICE в файле CONFIG.SYS. Команда внешняя. Файл SMARTDRV.EXE.

Формат команды:

Для запуска SMARTDRV из файла AUTOEXEC.BAT или из командной строки, формат команды SMARTDRV имеет вид:

диск:\маршрут\smartdrv *диск*+|-... /E:*размер_элем размер_кэша размер_кэша_Win* /B:*размер_буф* /C /R /L /Q /V /S

В том случае, если SMARTDRV уже загружен в память, формат команды имеет вид:

smartdrv *диск*+|-... /C /R

Параметры:

диск:\маршрут - указание местоположения файла SMARTDRV.EXE.

диск+|- - указание буквы диска для которого необходимо изменить параметры кэширования. Включив символ "+" в строку с обозначением буквы диска, пользователь разрешает кэширование диска для чтения и записи.

При использовании в строке символа "-" вместе с буквой диска отключается кэширование диска и для чтения и для записи. Если в командной строке SMARTDRV вы указываете букву диска без символа "+" или "-", то кэширование диска осуществляется только для чтения информации.

Если в команде SMARTDRV не указаны буквы дисков, то по умолчанию кэширование только для чтения включено для гибких дисков и дисков, созданных утилитой INTERLINK.

Для жестких дисков кэширование включено по умолчанию и на чтение и на запись, для CD-ROM дисков, сетевых дисков и Microsoft Flash Memory-Card дисков кэширование игнорируется.

Microsoft Flash Memory-Card - диски "мерцающей" памяти, выполненные в виде микросхем. Эти диски используют для своей организации новейшую технологию изготовления микросхем памяти (Flash-технология), позволяющую организовать на микросхемах памяти логические диски с сохранением информации после выключения питания, наподобие магнитных дисков.

/E:*размер_элем* - указание размера элемента кэша в БАЙТАХ. Допустимыми значениями являются следующие: 1024, 2048, 4096 и 8192. По умолчанию это значение равно 8192. Чем больше это значение, тем больше требуется памяти для работы утилиты SMARTDRV.

размер_кэша - указание размера кэша в КИЛОБАЙТАХ, который отводится при работе в DOS (когда Windows не запущена). Размер дискового кэша напрямую влияет на скорость выполнения дисковых операций. Чем больше кэш, тем реже SMARTDRV будет обращаться к диску для считывания отдельных фрагментов файлов. Если параметр *размер_кэша* не указан, то SMARDRV сам установит параметры кэширования. Размеры кэша, устанавливаемые SMARTDRV по умолчанию, приведены в таблице 8.2.

размер_кэша_Win - указание в килобайтах размера кэша, который резервируется для работы в среде Microsoft Windows. Когда вы выходите из Windows, размер кэша примет прежнее значение.

Размер кэша, резервируемый для Windows всегда меньше, чем размер кэша для работы в среде MS-DOS. Если параметр не указан, SMARTDRV сам установит размер кэша для Windows по умолчанию в соответствии с объемом оперативной памяти, установленной на компьютере.

/B:*размер_буф* - указание размера буфера для считанной информации. Буфер дополнительного чтения содержит информацию, которую SMARTDRV подгружает в память при загрузке прикладных программ. Для примера, если файл имеет длину 512K, он будет считан в память. При последующем обращении к файлу DOS будет осуществлять чтение информации не с диска, а из оперативной памяти, что значительно быстрее. По умолчанию размер буфера чтения равен 16 килобайт.

Следует помнить, что параметр *размер_буф* должен быть кратным параметру *размер_кэша*. Чем больше размер буфера чтения, тем больше памяти требуется для работы программы SMARTDRV;

Ключи

/C	запись всей информации из дискового кэша на кэшируемые диски. Обычно SMARTDRV записывает информацию на диски по мере необходимости. При использовании этого ключа информация будет записана на диск независимо от того, тре-

буется это или нет. Используйте этот ключ перед выключением компьютера или если вам необходимо, чтобы вся информация сразу же записывалась на диск. SMARTDRV записывает информацию на диск также в том случае, когда вы нажимаете клавиши Ctrl-Alt-Del для перезапуска вашей системы.

Помните, что для сохранения информации при перезапуске компьютера нельзя использовать кнопку выключения питания или сброса для перезагрузки компьютера, так как это приведет к потере данных;

/R очистка содержимого дискового кэша и рестарт SMARTDRV;

/L при использовании этого ключа SMARTDRV загружается в область основной памяти. При использовании SMARTDRV с двойной буферизацией необходимо добавить ключ /L в команду SMARTDRV.EXE вашего файла AUTOEXEC.BAT.

/Q краткая информация о команде SMARTDRV не будет выводится при его старте. Ключ /Q нельзя использовать совместно с ключом /V;

/V при использовании этого ключа на дисплей выводится текущий статус SMARTDRV и сообщения об ошибках при запуске утилиты. По умолчанию SMARTDRV не выводит на дисплей сообщения об ошибках. Ключ /V не допускается использовать совместно с ключом /Q;

/S организация вывода дополнительной информации о статусе SMARTDRV. В таблице 8.2 приведены значения параметров *размер_кэша* и *размер_кэша_Win*, которые устанавливаются по умолчанию SMARTDRV в зависимости от объема оперативной памяти.

Объем дополнительной памяти	Размер кэша	Размер кэша для Windows
До 1 MB	Вся XMS память	0 (нет кэширования)
До 2 MB	1 MB	256K
До 4 MB	1 MB	512K
До 6 MB	2 MB	1 MB
6 MB и более	2 MB	2 MB

Примеры использования:

Для создания дискового кэша с размером 256K (устанавливаемого по умолчанию для компьютера с ОЗУ до 2MB), необходимо ввести команду:

```
c:\dos\smartdrv
```

Для создания дискового кэша размером 2048 килобайт и резервирования кэша в 1024 килобайта для Windows, задайте команду SMARTDRV в виде:

```
c:\dos\smartdrv 2048 1024
```

Эту команду необходимо включить в состав вашего файла AUTOEXEC.BAT. Для разрешения кэширования операций чтения данных для гибких дисков и диска D: винчестера с одновременным запретом кэширования операций записи данных на гибких дисках необходимо ввести следующую команду в состав AUTOEXEC.BAT или задать ее из командной строки:

```
c:\dos\smartdrv A- B- D+
```

Для записи всей информации, накопленной в дисковом кэше на диск, необходимо ввести команду:

```
smartdrv /C
```

Примечания:

- Для работы SMARTDRV необходимо, чтобы компьютер имел дополнительную память и она должна быть доступна для работы DOS. Директива DEVICE файла CONFIG.SYS должна подключать драйвер управления XMS памятью. Причем, если вы подключаете SMARTDRV.EXE к системе в файле CONFIG.SYS директивой DEVICE, директива DEVICE, подключающая драйвер HIMEM.SYS и директива DOS=HIGH должны предшествовать подключению SMARTDRV.
- Драйвер SMARTDRV.EXE может быть загружен в блоки верхней памяти при условии, что они доступны ОС, командой LOADHIGH (LH). Подробности работы блоками старшей памяти рассматриваются в описании драйвера HIMEM.SYS - глава "Загружаемые драйверы устройств" и в описании команды LOADHIGH.
- Использование SMARTDRV для работы со "сжатыми" дисками не представляется особенно эффективным, так как функции оптимизации скорости работы с такими дисками в этом случае на себя берет утилита Double Space.

SORT

Сортировка строк информации, выводимой на экран, в файл или на другие устройства. Чаще всего утилита SORT работает как фильтр или конвейер, читающий символы в указанном столбце и преобразующий их в соответствии с указанным правилом. Команда внешняя. Файл SORT.EXE.

Форматы команды:

sort /R /+*n* < *диск1:\маршрут1\имя_файла1* > *диск2:\маршрут2\имя_файла2*

команда | sort /R /+*n* > *диск2:\маршрут2\имя_файла2*

Параметры:

диск1:\маршрут1\имя_файла1 - указание имени файла, данные которого поступают на вход команды SORT.

диск2:\маршрут\имя_файла2 - задание имени файла, в который будет записана выходная информация команды SORT.

команда - задание команды DOS, выходная информация которой будет направлена на вход утилиты SORT.

Ключи

/R	упорядочивание символов будет вестись в обратном порядке (от Z до A) и (от 9 до 0);
/+*n*	упорядочивание символов будет вестись, начиная с указанного параметром +*n* столбца. Если параметр не указан, утилита SORT осуществляет упорядочивание, начиная с 1-го столбца.

Примеры использования:

Допустим, вам необходимо отсортировать символы файла WORDS.TXT в обратном порядке. Для выполнения этой операции необходимо использовать команду SORT следующим образом:

```
sort /R < words.txt
```

В ходе выполнения этой команды файл WORDS.TXT будет считан с диска и символы его будут упорядочены в обратном порядке, начиная с 1-го столбца (по умолчанию). Очень часто команда SORT используется с командой FIND, как фильтр в операциях конвейеризации, что демонстрирует пример, приведенный ниже:

```
find "MS-DOS" document.txt | sort
```

В процессе выполнения блока из двух этих команд, в файле DOCUMENT.TXT осуществляется поиск фразы "MS-DOS" в строках текста и дальнейшая сортировка найденных строк командой SORT.

Команда SORT также может применяться и для сортировки текста, вводимого с клавиатуры при выводе его на дисплей. В этом варианте использования команды, достаточно ввести:

```
sort
```

После ввода команды пользователь может набирать текст, заканчивая ввод каждой набранной строки нажатием клавиши Enter. Для окончания набора текста и запуска утилиты SORT необходимо ввести код конца файла EOF (десятичный код 26). Это можно сделать либо при нажатии клавиш Ctrl-Z, либо задействовав клавишу F6. Нажав после ввода кода EOF клавишу Enter, вы укажете DOS на необходимость начала процесса сортировки только что введенного с клавиатуры текста. Далее DOS выведет на дисплей строки отсортированного текста.

Отсортированный текст можно сохранить в файле использовав при задании команды SORT символ ">>" операции перенаправления.

Примечания:

- Если в команде SORT не указано имя входного файла, то утилита SORT считает, что источником информации является стандартное для DOS устройство ввода-вывода информации - CON (консоль - дисплей и клавиатура). Для указания имени входного файла в команде SORT используется символ перенаправления ввода "<<" и символ конвейеризации "|". Для перенаправления выходной информации с дисплея необходимо использовать символ перенаправления вывода ">>".

  Для ускорения работы утилиты SORT в операциях перенаправления необходимо командой SET в файле AUTOEXEC.BAT установить окружение для переменной TEMP.
- При организации сортировки команда SORT использует текущие кодовые таблицы и национальные форматы специальных символов. При упорядочивании символов с кодами, большими кода ASCII 127 (десятич), коды символов и их изображение будут соответствовать текущим символам, информация о которых берется из файла COUNTRY.SYS. Для правильной установки кодовых страниц и национальных форматов специальных символов следует использовать директиву COUNTRY в файле CONFIG.SYS.
- При проведении сортировки текста команда SORT не обнаруживает различия между прописными и заглавными буквами.

- Утилита SORT не работает с файлами, длина которых превосходит предельную, которая равна 64 К.
- Для правильного использования команды SORT с другими фильтрами DOS (FIND и MORE) необходимо просмотреть описания этих команд и варианты их использования.

SUBST

Замена маршрута доступа буквой - идентификатором логического диска. Команда внешняя. Файл SUBST.EXE. Маршрут доступа к каталогу, описываемый буквой-идентификатором соответствующего логического диска и именами каталогов, получает обозначение нового логического диска.

Форматы команды:

subst *диск1: диск2:\маршрут*
subst *диск1:* /D

Для вывода на дисплей списка замененных маршрутов доступа (виртуальных дисков) необходимо задавать команду в следующем виде:

subst

Параметры:

диск1: - указание буквы виртуального диска, которой будет обозначаться измененный каталог.

диск2: - указание буквы, обозначающей физический диск, на котором расположен, подлежащий замене каталог (каталоги).

маршрут - указание каталога (списка каталогов), заменяемого на виртуальный диск.

Ключ

/D удаление виртуального диска.

Примеры использования:

Следующая команда создает виртуальный диск F: вместо маршрута доступа к файлам A:\LETTERS\JUL:

subst f: a:\letters\jul

После выполнения команды SUBST обращение к файлам каталога SUBST осуществляется через указание в параметрах команд только имени виртуального диска F:, например:

type f:jul25.93

Команда TYPE выводит содержимое файла JUL25.93 на экран с виртуального диска F:, но физически файл JUL25.93 находится в каталоге LETTERS\JUL логического диска A: . Для того, чтобы команда SUBST выполнилась без ошибок, необходимо в файле CONFIG.SYS правильно установить число максимально доступных ОС логических дисков директивой LASTDRIVE и, исходя из установок CONFIG.SYS, правильно задать букву виртуального диска.

Примечания:

- Команда SUBST создает логические диски, при работе с которыми имеется целый ряд ограничений. В частности, с дисками, созданными SUBST, не работают следующие команды DOS:

  ASSIGN, DISKCOMP, MIRROR, BACKUP, DISKCOPY, RESTORE, CHKDSK, FDISK, RECOVER, DATAMON, FORMAT, SYS, DEFRAG, LABEL.

- При попытке задания в качестве параметра *диск1:* буквы, выходящей из диапазона доступных букв, обозначающих логические диски (см. описание директивы LASTDRIVE файла CONFIG.SYS), на дисплее появится сообщение:

  Invalid parameter - drive:1
  (Неправильный параметр - диск:1)

- При работе в среде Microsoft Windows нельзя создавать или удалять виртуальные диски. Создание или удаление виртуальных дисков должно осуществляться либо до запуска Windows, либо после выхода из них.
- Команда SUBST может использоваться для создания виртуальных дисков вместо маршрутов доступа на "сжатых" дисках. Однако размер виртуального диска и свободного пространства не должны превышать аналогичных параметров для самого "сжатого" диска.

SYS

Перенос системных файлов MS-DOS (IO.SYS, MSDOS.SYS, DBLSPACE.BIN) и командного процессора MS-DOS (COMMAND.COM) на логический диск. После переноса файлов ОС командой SYS на диск он становится "системным" (загружаемым), то есть с него можно производить загрузку в память компьютера ОС после включения питания. Команда внешняя. Файл SYS.COM.

Формат команды:

sys *диск1:\маршрут диск2:*

Параметры:

диск1:\маршрут - указание местоположения "системных" файлов. Если этот параметр не указан, команда SYS осуществляет поиск системных файлов в корневом каталоге системного диска.

диск2: - указание буквы диска, на который необходимо перенести файлы ОС. Системные файлы могут быть скопированы командой SYS только в корневой каталог диска, в зарезервированное только для них место, и ни в какие другие каталог или место на диске.

Примеры использования:

Для копирования файлов ОС и командного процессора на диск A: необходимо ввести следующую команду:

```
sys a:
```

После выполнения этой команды диск A: станет "системным". Для переноса файлов ОС с диска A: на диск B: необходимо использовать команду SYS в следующем виде:

```
sys b: a:
```

Примечания:

- Если место на диске, отведенное под файлы MSDOS.SYS и IO.SYS, уже занято, не надо переформатировать диск. Начиная с версии 3.30 команда SYS сама освобождает место под файлы ОС, перемещая файлы, занимающие место файлов DOS в другие области диска.
- Команда SYS не позволяет переносить файлы DOS на диски, переназначенные командами ASSIGN, JOIN или SUBST. Команда SYS не работает также с сетевыми дисками и дисками, переназначенными утилитой INTERLNK.

TIME

Вывод на дисплей системного времени или его установка. Команда внутренняя. Информация о системном времени используется в указании времени создания каталогов, а также времени создания или изменения файлов.

Формат команды:

time *часы:минуты:секунды.сот_секунды* A|P

Для вывода на дисплей текущего системного времени, а также его изменения после запроса DOS необходимо использовать следующий формат команды:

```
time
```

Параметры:

часы - задание часов. Допустимое значение этого параметра лежит в диапазоне от 0 до 23.

минуты - указанием минут. Допускаются числа в диапазоне от 0 до 59.

секунды - указание секунд. Используются числа от 0 до 59.

сот_секунды - задание сотых долей секунды. Для указания этого параметра необходимо использовать числа от 0 до 99.

A|P - параметры используются для задания времени суток в формате, принятом в США. (12 часов до полудня A.M, и 12 часов после полудня P.M.). При указании параметров команды TIME используются целые положительные числа.

Примеры использования:

После установки системного времени вашего компьютера время может отображаться в одном из двух основных форматах (по умолчанию):

```
time 13:36
time 1:36p
```

Примечания:

- При попытке задания системного времени в неверном формате, MS-DOS выведет на дисплей сообщение в следующем виде:

  ```
  Invalid time
  (Неверное время)
  Enter new time:_
  (Введите новое время)
  ```

- Для отображении информации о системном времени DOS использует установки файла CONFIG.SYS (директива COUNTRY). В зависимости от кода страны DOS будет выводить информацию о системном времени в 12-часовом или 24-часовом формате. В том случае, если вы хотите изменить в данном сеансе работы системное время из 24-часового формата в 12-часовой формат, необходимо к указанию нового времени добавлять букву P или A.
- В том случае, если ваш компьютер не имеет встроенных часов, то в состав файла AUTOEXEC.BAT полезно включить команду TIME для установки времени системы пользователем каждый раз после запуска компьютера. Если же машина имеет встроенные часы, питающееся от батарейки, то система сама правильно установит системное время, соответствующее текущему времени суток. В противном случае, после запуска компьютера отсчет системного времени осуществляется от 00:00:00.00.

TREE

Вывод на экран дерева каталогов.

Команда внешняя. Файл TREE.COM.

Формат команды:

tree *диск:\маршрут* /F /A

Параметры:

диск: - указание буквы логического диска, структуру каталогов которого необходимо вывести.

маршрут - указание каталога, структуру подкаталогов которого необходимо вывести на дисплей.

Ключи

/F	вывод на дисплей списка файлов каждого каталога;
/A	при указании этого ключа для построения структуры каталогов будет использовать только текстовые символы (не включая символы псевдографики). Ключ используется в тех случаях, когда текущая кодовая таблица не поддерживает символы псевдографики или при выводе на принтер структуры каталогов происходит искажение рисунка, видимого на экране дисплея.

Примеры использования:

Для получения на экране дисплея структуры каталогов текущего диска в графическом виде, введите команду:

```
tree \
```

Команду TREE можно использовать совместно с другими командами DOS в операциях перенаправления, например, вместе с командой MORE для постраничного вывода на дисплей выходной информации команды TREE. Нижеприведенный пример показывает, как использовать утилиту TREE совместно с командой MORE:

```
tree e:\ /F | more
```

Выходная информация команды TREE может быть сохранена в указанном файле или передана на вход устройства отображения информации (например на принтер PRN). Следующий пример показывает, как сохранить выходную информацию TREE в файле с именем TREE.RPT

```
tree e:\ /F  tree.rpt
```

Примечания:

- В том случае, когда в команде TREE не указаны параметры *диск:* и *маршрут*, считается, что необходимо вывести структуру подкаталогов (и файлов) текущего каталога.
- Информация о структуре каталогов и файлов может быть также получена с помощью команды DIR. Подробности работы с командой DIR приведены в ее описании.

TYPE

Вывод на дисплей содержимого текстового файла. Использование команды TYPE приводит к просмотру текстового файла без его изменения. Команда внутренняя.

Формат команды:

type *диск:\маршрут\имя_файла*

Параметр:

диск:\маршрут\имя_файла - указание полного идентификатора файла, содержимое которого необходимо просмотреть.

Примеры использования:

Для просмотра содержимого файла BIRTHDAY.TXT, расположенного в текущем каталоге, введите команду:

```
type birthday.txt
```

Очень удобно использовать команду TYPE совместно с командой MORE при выводе на дисплей файлов, содержимое которых не умещается на экране. Использование команды MORE в операциях конвейеризации позволит организовать постраничный вывод информации.

```
type birthday.txt | more
```

Примечание:

- Для лучшего использования команды TYPE в операциях перенаправления установите командой SET окружение для переменной TEMP в файле AUTOEXEC.BAT.
- При выводе командой TYPE содержимого двоичных файлов, на экране могут появляться странные символы. Этот эффект связан с наличием в двоичном (бинарном) файле специальных управляющих кодов, например, кодов перевода формата, возврата каретки, которые обычно отсутствуют в текстовых файлах. Следует избегать использования команды TYPE для отображения на дисплее содержимого двоичных файлов.

UNDELETE

Восстановление файлов, ранее удаленных командой DEL. Утилита UNDELETE предлагает восстановление случайно удаленных файлов при помощи методов *дежурного удаления, поискового удаления и стандартного удаления*. Подробно каждый уровень защиты файлов обсуждается в примечаниях к этой команде. Команда внешняя. Файл UNDELETE.EXE.

Форматы команды:

undelete *диск:\маршрут\имя_файла* /DT|/DS|/DOS
undelete /LIST|/ALL|/PURGE*диск*|/STATUS|/LOAD|/UNLOAD /S*диск*|/T*диск - число*

Параметры:

диск:\маршрут\имя_файла - указание полного имени файла (файлов), которые необходимо восстановить. По умолчанию UNDELETE восстанавливает все файлы текущего каталога.

Ключи

/LIST — вывод на дисплей списка удаленных файлов, которые можно восстановить. В список не будут включаться файлы, которые невозможно восстановить. Параметры *диск:\маршрут\имя_файла* и ключи /DT, /DS, /DOS автоматически включают этот ключ в команду UNDELETE. Ключ /LIST вместе со списком удаленных файлов показывает также и степень их повреждения;

/ALL — восстановление файлов с запросом у пользователя разрешения на восстановление каждого файла. Для восстановления файлов UNDELETE первоначально восстановить файлы дежурного удаления - "Delete Sentry".

Если восстановление удаленных файлов по указанному методу невозможно, тогда используется метод поискового восстановления файлов - "Delete Tracker".

Принцип восстановления утилитой UNDELETE удаленных файлов заключается в поиске в каталогах (корневом и подчиненных ему) файлов, имя которых начинается с символа "#". Файлы с такими именами DOS считает удаленными. При восстановлении таких файлов DOS попросит ввести первый

символ имени восстанавливаемого файла. Если на диске уже существует в том каталоге, где будет восстановлен файл, другой файл с таким же именем, то DOS попросит ввести новую букву для имени диска. Файл не будет восстановлен до тех пор, пока его имя не будет уникальным среди файлов каталога, куда будет восстановлен файл. Для указания первого символа имени файла следует использовать только символы, разрешенные для задания имен файлов;

/DOS — при восстановлении файлов будет использоваться информация из каталогов с запросом у пользователя разрешения на восстановление каждого файла;

/DT — при восстановлении файлов будет использоваться информация из файла трассировки. Программа UNDELETE будет запрашивать разрешение у пользователя на восстановление каждого файла;

/DS — при восстановлении файлов будет использоваться информация из дежурного каталога "SENTRY" с запросом подтверждения на восстановление каждого файла;

/LOAD — загрузка в память программы UNDELETE с последующим ее использованием как резидентной программы. При загрузке в память программа UNDELETE использует установки, заданные в файле UNDELETE.INI. В том случае, если файл UNDELETE.INI не существует, утилита UNDELETE будет использовать установки по умолчанию;

/UNLOAD — при использовании этого ключа из оперативной памяти будет выгружена резидентная часть программы UNDELETE. Эта операция делает невозможным осуществление постоянного контроля за удаленными файлами;

/PURGE*диск:* — удаление содержимого дежурного каталога с именем SENTRY. Если параметр *диск:* в ключе не указан, утилита UNDELETE осуществляет поиск каталога с именем SENTRY на текущем диске;

/STATUS — вывод на дисплей типа защиты от случайного удаления файлов, примененной на каждом диске;

/S*диск* — включение метода защиты файлов с использованием дежурного каталога - "Delete Sentry" и загрузка

резидентной части программы UNDELETE в память. При использовании этого ключа информация об удаленных файлах указанного диска записывается в в каталог с именем SENTRY. Если в ключе /S параметр *диск* не указан, то происходит сохранение информации об удаленных файлах текущего диска. При загрузке программы UNDELETE используется информация о ее установках, определенная в файле UNDELETE.INI.

/T*диск - число* указание *числа* файлов логического *диска*, о которых программа UNDELETE будет сохранять информацию в случае их удаления. При использовании этого ключа производится одновременная загрузка резидентной части программы UNDELETE в оперативную память машины. Указание этого ключа приводит к использованию поисковой системы защиты от копирования - "Delete Tracker". Параметр *число* может принимать значения от 1 до 999. Информация об удаленных файлах, при использовании этого ключа сохраняется в файле с именем PCTRACKR.DEL. Если при использовании ключа параметр *число* не указан, то утилита UNDELETE сама устанавливает число файлов, о которых программа будет хранить информацию, в соответствии с таблицей 8.3.

Таблица 8.3

Число удаленных файлов, о которых UNDELETE (по умолчанию) будет хранить информацию

Размер диска	Число файлов	Размер файла PCTTRACKR.DEL
360K	25	5K
720K	50	9K
1.2 MB	75	14K
1.44 MB	75	14K
20 MB	101	18K
32 MB	202	36K
32 MB	303	55K

Примеры использования:

Для загрузки программы UNDELETE в память вашего компьютера и восстановления всех удаленных файлов в текущем каталоге текущего диска необходимо ввести следующую команду.

```
undelete
```

При обнаружении каждого удаленного файла UNDELETE будет запрашивать подтверждения на его восстановление.

Для восстановления всех файлов из корневого каталога диска B:, имеющих расширение .TXT с использованием информации из корневого каталога диска B:, необходимо ввести следующую команду:

```
undelete b:\*.txt /DOS
```

Для восстановления файлов с использованием скрытого каталога SENTRY и загрузки в память резидентной части программы UNDELETE необходимо ввести следующую команду DOS:

```
undelete /SD
```

Приведенная выше команда позволит сохранять информацию об удаленных файлах на диске D:, в скрытом каталоге с именем SENTRY.

```
undelete /Te-600
```

При выполнении этой команды на диске E: для защиты от случайного удаления файлов будет использован метод поискового удаления с сохранением в специальном файле с именем PCTRACTR.DEL информации о 600 удаленных файлах. Команда восстановления файла с использованием файла трассировки:

```
undelete test.exe /DT
```

Примечания:

Метод защиты файлов от удаления "Delete Tracker" нельзя использовать для дисков, переназначенных командами JOIN или SUBST.

При использовании команды ASSIGN перед запуском этой утилиты вы должны запустить утилиту UNDELETE с использованием защиты файлов методом поискового восстановления файлов.

После того, как вы удалили нужный вам файл, и его необходимо восстановить, не предпринимайте никаких действий с файлами (операции копирования, перемещения, создания файлов и т.д.), так как это может привести к использованию другими файлами кластеров удаленного файла, и невозможности восстановления удаленного файла.

Утилита MS-DOS 6.0 UNDELETE предлагает пользователям совершенно новые для DOS уровни защиты файлов от случайного удаления:

- Дежурное удаление - "Delete Sentry". Требует для своей работы минимум оперативной памяти и максимум дискового пространства.
- Поисковое удаление - "Delete Tracker". Требует минимум дискового пространства.
- Стандартная защита - "Standard". Низший уровень защиты. Требует минимум оперативной памяти и не занимает дискового пространства.

Рассмотрим подробно работу каждого метода защиты от удаления файлов.

ДЕЖУРНОЕ УДАЛЕНИЕ - "DELETE SENTRY"

При использовании этого метода утилита UNDELETE создает скрытый каталог SENTRY, в котором хранит информацию об удаленных файлах. Когда пользователь производит удаление файла, утилита SENTRY производит перемещение файла из текущего каталога в каталог с именем SENTRY без удаления имени файла из таблицы размещения файлов FAT.

В процессе восстановления файлов утилита UNDELETE перемещает восстанавливаемый файл из каталога с именем SENTRY в каталог, где находился до этого восстанавливаемый файл. Размер файлов, входящих в каталог SENTRY, ограничен и составляет примерно 7 % от общей емкости диска.

Если каталог SENTRY не может вместить последний удаленный файл, то утилита UNDELETE производит удаление "ранних" удаленных файлов до тех пор, пока места в каталоге SENTRY не будет достаточно для размещения последнего удаленного файла.

При использовании метода "Delete Sentry" для работы утилиты UNDELETE необходимо иметь свободными в оперативной памяти 13.5 K, где размещается резидентная часть программы UNDELETE.

ПОИСКОВОЕ УДАЛЕНИЕ - "DELETE TRACKER"

Поисковое удаление является более низким по степени восстановления файлов методом защиты от случайного удаления. При использовании этого метода на указанном диске создается "скрытый" файл с именем PCTRACKER.DEL, содержащий записи об удаленных файлах.

При удалении файла MS-DOS изменяет записи таблицы размещения файлов FAT для сообщения системе, что кластеры, занимаемые файлом, свободны для записи других файлов (подробно вопрос работы DOS с FAT обсуждается в главе "Файловая система"). Восстановление файла при использовании этого метода заключается в сообщении другому файлу, что место, выделяемое ему, уже занято. Если же другой файл

все-таки займет это место, то в этом случае возможно лишь частичное восстановление файлов.

Для работы UNDELETE в режиме "поискового" удаления необходимо 13.5 K оперативной памяти (резидентная часть UNDELETE) и минимум места на диске для размещения файла PCTRACKER.DEL.

СТАНДАРТНЫЙ УРОВЕНЬ - "STANDARD"

Стандартный уровень защиты от случайного удаления файлов доступен пользователю сразу после загрузки компьютера. Стандартный уровень защиты является низшим уровнем защиты от удаления файлов. Этот метод не требует дополнительных ресурсов памяти и дискового пространства, как два ранее описанных метода. Однако он, по сравнению с другими методами, обеспечивает и наименее надежное восстановление удаленных файлов.

Утилита UNDELETE не позволяет восстановить каталог, который был перемещен, и не может восстановить файлы, который находились в перемещенном каталоге.

Если восстанавливаемый каталог является каталогом, подчиненным непосредственно корневому каталогу, используйте сначала команду UNFORMAT для восстановления каталога, а затем утилиту UNDELETE для восстановления файлов.

При запуске команды UNDELETE утилита использует файл UNDELETE.INI, содержащий текущие установки программы. Когда файл UNDELETE.INI не существует, программа UNDELETE создаст его. Файл UNDELETE.INI содержит следующие пункты:

- "дежурные" диски;
- "дежурные" файлы;
- "зеркальные" диски;
- конфигурация;
- установки по умолчанию.

Раздел "дежурные" диски указывает на диски, защищенные с использованием метода "Delete Sentry".

В разделе "дежурные" файлы приведен список файлов, защищенных с использованием методов "Delete Tracker" или "Delete Sentry". Наличие символа "-" рядом с именем файла показывает файлы, которые не записаны.

По умолчанию этот раздел файла PCTRACKER.DEL имеет вид:

. -*.TMP -*.VM? -*.WOA -*.SWP -*.SPL -*.RMG -*.IMG -*.THM -*.DOV

В разделе "зеркальные" файлы приведены сведения о дисках, защищенных методом "Delete Tracker".

Раздел конфигурация содержит сведения по следующим пунктам:

- включена ли защита от удаления для "архивных" файлов. По умолчанию для "архивных" файлов такой защиты нет и в разделе "конфигурация" имеется строка:

```
archive = FALSE ;
```

- число дней, в течение которых будет сохраняться информация о файлах. По умолчанию число дней равно 7, и в разделе "конфигурация" будет иметься запись:

```
days = 7;
```

- количество зарезервированной дисковой поверхности для хранения информации об удаленных файлах. По умолчанию этот параметр равен 20, и в разделе "конфигурация" имеется строка:

```
persentage = 20;
```

- установки методов защиты файлов от удаления по умолчанию. Для указания методов защиты в файле PCTRACKER.DEL имеются строки:

```
[defaults]
d.sentry = TRUE
d.tracker = FALSE
```

Программа UNDELETE защищает от удаления файлы всех дисков, указанных в файле UNDELETE.INI, включая диски, указанные при старте программы. Допустим, файл UNDELETE.INI содержит следующие строки:

```
[sentry.drives]
D=
E=
```

Следующая команда включит метод защиты "Delete Sentry" для обоих дисков D: и E::

```
undelete /SD
```

Для изменения метода защиты файлов, используемого UNDELETE, необходимо выполнить следующие действия:

1. Необходимо выгрузить из памяти резидентную часть утилиты UNDELETE командой UNDELETE /U.

2. Установить новый уровень защиты и перезагрузить резидентную часть программы UNDELETE. Для примера, установим уровень защи-

ты "Delete Tracker" для диска D: с запоминанием информации о 200 удаленных файлов:

```
undelete /Td-200
```

Если вам необходимо изменить установки UNDELETE, задаваемые утилите при ее запуске, необходимо изменить соответствующие разделы файла UNDELETE.INI.

Для защиты файлов на сетевых дисках следует использовать уровень защиты "Delete Sentry" и сделать возможным чтение, запись, создание и удаление файлов только в корневом каталоге соответствующего диска.

UNFORMAT

Восстановление случайно отформатированного командой FORMAT диска. Команда может применяться для восстановления всех локальных жестких и логических дисков и не может использоваться для восстановления сетевых дисков. При своей работе команда UNFORMAT может также восстановить испорченную таблицу разделов на жестком диске.

Команда внешняя. Файл UNFORMAT.COM.

Формат команды:

unformat *диск:* /L /TEST /P

Параметры:

диск: - указание имени диска, файловую структуру которого необходимо восстановить.

Ключи

/L	вывод на дисплей информации о файлах и каталогах по мере расформатирования диска. Если ключ /L не используется в команде UNFORMAT, то выводится информация только о тех файлах, которые нельзя восстановить без помощи пользователя. Для приостановки вывода списка файлов при расформатировании диска необходимо использовать комбинацию клавиш Ctrl-S. Для продолжения просмотра необходимо нажать любую клавишу;
/TEST	вывод утилитой информации о том, как будет восстановлено содержимое диска. При вводе команды UNFORMAT с этим ключом восстановление диска не производится;
/P	перенаправление выходной информации команды UNDELETE на принтер, подсоединенный к порту LPT1.

Пример использования:

Для определения того, как команда UNFORMAT планирует восстановить отформатированный диск B:, необходимо ввести следующую команду:

```
unformat b: /TEST
```

Если вам необходимо восстановить случайно отформатированный диск D: с показом списка файлов и подкаталогов по мере их расформатирования, используйте команду UNFORMAT в следующем виде:

```
unformat d: /L
```

Примечания:

- Утилита UNDELETE не позволяет восстановить диски, отформатированные командой FORMAT с ключом /U. Подробности работы команды FORMAT с этим ключом приведены в ее описании.
- При своей работе утилита UNFORMAT выводит на дисплей список всех подкаталогов, которые она нашла. При использовании ключа /L пользователь имеет возможность просмотра списка файлов, входящих в каждый найденный каталог.
- В процессе работы UNFORMAT пытается восстановить на логическом диске корневой каталог и таблицу размещения файлов.
- При обнаружении утилитой UNFORMAT фрагментированных файлов, она пытается найти все кластеры, которые файл занимает. Если же отдельные кластеры файла не будут найдены, или они будут заняты другими файлами, то восстанавливаемый файл усекается или совсем удаляется.
- Работа утилиты UNFORMAT возможна только с дисками, имеющими размер сектора 512, 1024 или 2048 байт.

VER

Выдача на дисплей номера версии операционной системы MS-DOS. Команда внутренняя.

Формат команды:

ver

Пример использования:

После ввода команды VER MS-DOS выведет на дисплей сообщение следующего характера:

```
MS-DOS Version 6.0
```

VERIFY

Включение или выключение проверки правильности выполнения дисковых операций чтении-записи информации.

Эта команда обычно используется в тех случаях, когда необходимо гарантировать сохранность информации при записи ее на диск. Используя эту команду, например, вы никогда не запишете данные в плохой сектор на диске, который не обнаружен другими средствами DOS.

Команда может быть задана как в файле CONFIG.SYS, так и из командной строки.

Команда внутренняя.

Формат команды:

verify ON|OFF

Ключи

ON|OFF	включение-выключение верификации (проверки) правильности выполнения дисковых операций. По умолчанию ключ установлен в положение OFF.

Примечания:

- Для вывода на дисплей текущего статуса команды VERIFY задайте ее без параметров.
- Использование команды VERIFY с ключом ON повышает надежность выполнения дисковых операций, но увеличивает время записи-чтения файлов (порой довольно значительно).
- Для получения информации о количестве плохих секторов на диске можно использовать команду CHKDSK, либо прикладную программу из дисковых инструментальных пакетов, например утилиту Norton Disk Doctor из пакета Norton Utilites.
- Включение верификации возможно и при использовании следующих команд DOS: COPY и XCOPY с ключом /V. Подробности использования этих команд даны в их описании.

VOL

Вывод на дисплей метки тома и серийного номера указанного диска, если он существует. Команда внутренняя.

Формат команды:

vol *диск:*

Параметр:

диск: - указание имени диска, метку тома которого и серийный номер необходимо вывести на дисплей.

Примечания:

Если вы задаете команду VOL без указания буквы диска, на дисплей будет выведена метка текущего диска.

(!) VSAFE

Проверка вашего компьютера на вирусы и выдача предупреждения пользователю при обнаружении VSAFE нестандартной работы машины. VSAFE является резидентной программой и занимает для работы 22K оперативной памяти. Команда внешняя. Файл VSAFE.COM.

Формат команды:

vsafe /*установка*+|-... /NE /NX /A*x*|/C*x* /N /D /U

Параметры:

установка - этот параметр указывает VSAFE, как и против каких вирусов должен реагировать VSAFE. Используйте символы "-" и "+" для выключения или включения соответствующей установки утилиты. Список установок VSAFE и функции, ими реализуемые, приведены ниже:

1 выдача предупреждения пользователю при попытке форматирования жесткого диска на низком уровне - "Low Level Format". По умолчанию ключ включен;

2 при включенном положении этого ключа программа загружается и остается резидентной. По умолчанию ключ находится в положении "выключено";

3 включение запрещения записи всех файлов на диск. По умолчанию ключ выключен;

4 проверка выполняемых файлов при их открытии MS-DOS. По умолчанию ключ установлен в положение "включено";

5 проверка стартовых секторов "boot sectors" всех дисков на вирусы. По умолчанию этот ключ задействован;

6 включение защиты от вирусов для стартового сектора и таблицы разделов жесткого диска (дисков). По умолчанию этот ключ установлен в положение "включено";

7 включение защиты от вирусов для стартового сектора гибкого диска (дисков). По умолчанию этот ключ установлен в положение "выключено";

8 включение защиты для выполняемых файлов. По умолчанию - защита выключена.

Ключи

/NE	при использовании этого ключа программа VSAFE будет загружена в область расширенной памяти;
/NX	загрузка программы VSAFE в область дополнительной памяти;
/A*x*	установка "горячего" ключа для вызова программы с использованием клавиши Alt и клавиши с обозначением символа, указанного параметром *x*;
/C*x*	установка "горячего" ключа для вызова программы с использованием клавиши Ctrl и клавиши с обозначением символа, указанного параметром *x*;
/N	при задании этого ключа VSAFE будет обнаруживать все знакомые ему вирусы и на сетевых дисках;
/D	выключение проверки контрольных сумм файлов;
/U	удаление резидентной части программы VSAFE из оперативной памяти.

Управлять программой VSAFE можно также и после ее запуска, вызвав ее меню в любой доступной VSAFE программе через нажатие "горячих" клавиш. "Горячие" клавиши устанавливаются ключами /A*x* и /C*x*. По умолчанию "горячей" клавишей является комбинация Alt-V. Вид программного меню VSAFE приведен на рисунке 8.11.

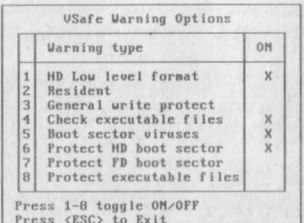


Рис. 8.11

Пользователь может изменять установки утилиты VSAFE, нажимая клавиши от 1 до 8. В соответствующей строке меню при наборе соответствующей установки будет появляться символ "X". Внизу меню отображается подсказка пользователю по работе с VSAFE.

Пример использования:

Для включения зашиты Boot-секторов гибких дисков и выполняемых файлов, а также установки для входа в меню программы "горячего" ключа Ctrl-R, используйте команду VSAFE в следующем виде:

```
vsafe /7+ /8+ /Cr
```

Примечания:

- Перед инсталляцией Microsoft Windows необходимо утилиту VSAFE выгрузить из памяти, так как она будет мешать нормальному процессу инсталляции Windows.
- Для запуска в среде Microsoft Windows программы VSAFE (элемент MWAVTSR) необходимо перед запуском Windows запустить VSAFE в среде MS-DOS.
- Если вам необходимо использовать VSAFE в среде Windows как резидентную антивирусную программу, добавьте сами в файл WIN.INI строку:

  ```
  load = mwavtsr.exe
  ```

- Имеющиеся в версии MS-DOS 6.0 встроенные утилиты для Windows кратко обсуждаются в Приложениях для этой книги.

XCOPY

Выборочное копирование групп файлов, каталогов и подкаталогов (исключая "скрытые" и "системные" файлы).

Используя эту команду, можно осуществить выборочное копирование файлов из указанного каталога, включая также файлы во всех подчиненных указанному подкаталогах. Команда внешняя. Файл XCOPY.EXE.

Формат команды:

xcopy *источник приемник* /A | /M /D:*дата* /P /S /E /V /W

Параметры:

источник - указание местоположения на дисках и имен файлов, подлежащих копированию. Параметр *источник* должен включать в себя указание диска и маршрута доступа к копируемым файлам.

приемник - задание диска и каталога, в который будет осуществлено копирование файлов.

Ключи

/A — копируются файлы, у которых установлен атрибут "архивный". Такой атрибут обычно имеют модифицированные или вновь созданные файлы. Дополнительная информация об атрибутах файлов приводится в описании команды ATTRIB. Копирование файлов командой XCOPY не изменяет флага архивации;

/M — копируются файлы, у которых установлен флаг архивации. Команда XCOPY с ключом /M сбрасывает флаг архивации, в отличие от ключа /A, который не выполняет эту операцию;

/D:*дата* — копируются только те файлы, которые модифицированы или созданы в указанный день, либо после него. Формат даты определяется установкой директивы COUNTRY файла CONFIG.SYS;

/S — при использовании этого ключа копируются файлы из всех каталогов, подчиненных указанному, если они не пусты. В том случае, когда ключ опущен, копируются файлы только из указанного каталога;

/P — перед копированием каждого файла DOS запрашивает у пользователя разрешение на проведение операции копирования;

/V — этот ключ задает режим верификации при проведении операций копирования файлов;

/E — при использовании этого ключа на диске, обозначенном параметром *приемник*, создаются копии каталогов *диска-источника*, даже в том случае, если они являются пустыми;

/W — перед копированием каждого файла DOS выводит на дисплей запрос и ожидает установки в дисковод дискеты с файлами, предназначенными для копирования. При использовании этого ключа перед началом копирования на дисплее отобразится сообщение:

Press any key to begin copying file(s)
(Нажми любую клавишу для копирования файла(ов))

Для прерывания процесса копирования файлов необходимо использовать комбинацию клавиш Ctrl-C.

Примеры использования:

Для копирования всех файлов с диска C: на диск D:, включая все файлы во всех подчиненных корневому каталогах, включая даже пустые каталоги, введите команду XCOPY в следующем виде:

```
xcopy c:\*.* d: /S /E
```

Следующий пример иллюстрирует копирование файлов из части дерева каталогов, начиная с каталога \BIN\UNITS, и только тех файлов, которые созданы, начиная с 29.09.93 и позже:

```
xcopy c:\bin\units\ d: /D:29/09/93 /S /E /V
```

В приведенной выше команде использован ключ /V, который задает режим верификации при проведении операций копирования файлов.

Примечания:

- Когда в команде XCOPY вы опускаете указание параметра "приемник", команда XCOPY производит копирование файлов в текущий каталог.
- Команда XCOPY не позволяет копировать "скрытые" и "системные" файлы. Для изменения атрибутов файлов используйте команду ATTRIB.
- В том случае, если в параметре "приемник" вы не разделяете имя каталогов символом "\", то утилита XCOPY выведет на дисплей следующее сообщение:

  ```
  Does destination specify a file name
  or directory name on the target
  (F = file, D = directory)?
  (Не указано имя файла в приемнике)
  (или имя каталога назначения)
  ((F = файл, D = каталог)?)
  ```

  Нажмите клавишу F, если вы хотите, чтобы файл (файлы) были скопированы в файл. Нажмите клавишу D для копирования файлов в каталог.

- В том случае, если вы хотите сделать копию диска со структурой размещения файлов, отличающейся от структуры на диске-оригинале, то используйте для этих целей команду XCOPY. Такая ситуация может возникнуть в случае, когда файлы на диске-оригинале фрагментированы, а на диске-копии необходимо иметь нефрагментированные файлы. В том случае, когда вам нужно создать полную копию диска-оригинала, например, при наличии защищенной дискеты, используйте команду DISKCOPY.

- По окончании своей работы XCOPY возвращает MS-DOS следующие коды своего завершения:

0 файлы скопированы успешно;
1 файлов, предназначенных для копирования, не обнаружено;
2 пользователь использовал для прекращения копирования комбинацию клавиш Ctrl-C;
4 обнаружена ошибка инициализации. Такая ошибка возникает в случае, когда утилите XCOPY недостаточно оперативной памяти, дискового пространства, указано неправильное имя диска или использован неверный формат команды;
5 обнаружена ошибка записи на диск;

Коды завершения, возвращаемые командой XCOPY, могут в дальнейшем быть использованы другими прикладными программами, но чаще всего они используются в командных файлах (команда IF ERRORLEVEL).

9 Приложения

ПРИЛОЖЕНИЕ 1. Сводный список команд MS-DOS по версиям.

Команда	Версия MS-DOS			
	3.30	4.0	5.0	6.0
APPEND	+	!	+	+
ASSIGN	+	+	+	-
ATTRIB	+	!	!	+
BACKUP	+	+	!	-
BREAK	+	+	+	+
CHCP	+	+	+	+
CHDIR (CD)	+	+	+	+
CHKDSK	+	+	+	+
CLS	+	+	+	+
COMMAND	+	+	+	!
COMP	+	!	!	-
COPY	+	+	+	+
CTTY	+	+	+	+
DATE	+	+	+	+
DBLSPACE	-	-	-	+
DEBUG	+	!	+	+
DEFRAG	-	-	-	+
DEL	+	+	+	+
DELTREE	-	-	-	+
DIR	+	+	!	!
DISKCOMP	+	+	+	+

Команда	Версия MS-DOS			
	3.30	4.0	5.0	6.0
DISKCOPY	+	+	!	+
DOSKEY	-	-	+	!
DOSSHELL	-	+	!	+
EDIT	-	-	-	+
EMM386	-	-	+	+
ERASE	+	!	+	+
EXE2BIN	+	+	+	-
EXIT	+	+	+	+
EXPAND	-	+	+	+
FASTHELP	-	-	-	+
FASTOPEN	+	!	+	!
FC	-	+	+	+
FDISK	+	!	+	+
FIND	+	+	!	+
FORMAT	+	!	!	+
GRAPHTABL	+	!	+	-
GRAPHICS	+	+	+	!
HELP	-	-	+	!
JOIN	+	+	+	-
INTERLNK	-	-	-	+
INTERSVR	-	-	-	+
KEYB	+	+	+	!
LABEL	+	+	+	+
LOADFIX	-	-	-	+
LOADHIGH (LH)	-	-	+	!
MEM	-	+	+	!
MEMMAKER	-	-	-	+
MIRROR	-	-	+	-
MKDIR (MD)	+	+	+	+
MODE	+	!	+	+
MORE	+	+	+	+
MOVE	-	-	-	+
MSAV	-	-	-	+
MSBACKUP	-	-	-	+
MSCDEX	-	-	-	+

Команда	Версия MS-DOS			
	3.30	4.0	5.0	6.0
MSD	-	-	-	+
NLSFUNC	+	+	+	+
PATH	+	+	+	+
POWER	-	-	-	+
PRINT	+	+	+	+
PROMPT	+	+	+	+
RECOVER	+	+	+	-
RENAME (REN)	+	+	+	+
REPLACE	+	+	+	+
RESTORE	+	+	+	+
RMDIR (RD)	+	+	+	+
SET	+	+	+	+
SETVER	-	-	+	+
SHARE	+	!	+	+
SMARTDRV	-	-	-	+
SORT	+	+	+	+
SUBST	+	+	+	+
SYS	+	+	+	+
TIME	+	+	+	+
TREE	+	!	+	+
TYPE	+	+	+	+
UNDELETE	-	-	+	!
UNFORMAT	-	-	+	!
VER	+	+	+	+
VERIFY	+	+	+	+
VOL	+	+	+	+
VSAFE	-	-	-	+
XCOPY	+	+	+	+

Условные обозначения:

- \- отсутствие команды в версии DOS;
- ! наличие в команде дополнительных возможностей по сравнению с предыдущими версиями DOS;
- \+ присутствие команды в версии DOS.

ПРИЛОЖЕНИЕ 2. Список внутренних команд DOS.

BREAK	DATE	MKDIR (MD)	TIME
CHCP	DEL	PATH	TYPE
CHDIR (CD)	DIR	PROMPT	VER
CLS	ERASE	RENAME (REN)	VERIFY
CTTY	EXIT	RMDIR (RD)	VOL
COPY	LOADHIGH (LH)	SET	

ПРИЛОЖЕНИЕ 3. Esc-последовательности драйвера ANSI.SYS.

Esc[2J	очистка экрана и перемещение курсора в левый верхний угол экрана.
Esc[K	очистка строки от курсора и до конца строки.
Esc[стр.,кол.H	перемещение курсора в указанную строку (параметр стр., который меняется в пределах от 1 до 25 для режима 80x25), в указанный параметром кол. столбец (параметр кол. меняется в пределах от 1 до 80, режим 80x25). Символ "H" может быть заменен символом "f".
Esc[числоA	перемещение курсора на число строк вверх.
Esc[числоB	перемещение курсора на число строк вниз.
Esc[числоC	перемещение курсора на число позиций вправо.
Esc[числоD	перемещение курсора на число позиций влево.
Esc[s	сохранение текущих координат курсора в специальном буфере.
Esc[u	восстановление координат курсора, сохраненных в специальном буфере.
Esc[6n	вывод текущих координат курсора в специальном формате Esc[стр.,кол.R
Esc[=7n	включение автоматического перевода курсора на следующую строку (автоматическое добавление символа CR в конец строки ввода).

Esc[=7l — выключение автоматического перевода курсора на следующую строку.

Esc[=числоh — выбор режима работы дисплея. Вместо символа h допускается использование символа l.

Параметр "число" может принимать следующие значения :

- 0 - 40 x 25 монохромный режим (текст);
- 1 - 40 x 25 цветной режим (текст);
- 2 - 80 x 25 монохромный режим (текст);
- 3 - 80 x 25 цветной режим (текст);
- 4 - 320 x 200 4-цветный режим (графика);
- 5 - 320 x 200 монохромный режим (графика);
- 6 - 640 x 200 монохромный режим (графика);
- 7 - разрешение "заворачивания" изображения;
- 13 - 320 x 200 цветной режим (графика);
- 14 - 640 x 200 цветной режим (16-ти цветная графика);
- 15 - 640 x 350 монохромный режим (2-х цветная графика);
- 16 - 640 x 350 цветной режим (16-ти цветная графика);
- 17 - 640 x 480 монохромный режим (2 - х цветная графика);
- 18 - 640 x 480 цветной режим (16-ти цветная графика);
- 19 - 320 x 200 цветной режим (256-ти цветная графика);

Esc[числоm - выбор цветовых атрибутов для фона и текста.

Параметр "число" в этой команде может принимать следующие значения :

- 0 - нормальное изображение (белые символы на черном фоне);
- 1 - выделение яркостью;
- 4 - выделение подчеркиванием (режим доступен только на монохромных дисплеях);
- 5 - выделение мерцанием символов;

- 7 - инверсное изображение;

- 8 - скрытый текст (только для монохромных дисплеев).

Для установки цвета символов и фона используйте значения :

- 30 - черные символы;
- 31 - красные символы;
- 32 - зеленые символы;
- 33 - желтые символы;
- 34 - синие символы;
- 35 - фиолетовые символы;
- 36 - бирюзовые символы;
- 37 - белые символы;
- 40 - черный фон;
- 41 - красный фон;
- 42 - зеленый фон;
- 43 - желтый фон;
- 44 - синий фон;
- 45 - фиолетовый фон;
- 46 - бирюзовый фон;
- 47 - белый фон.

ПРИЛОЖЕНИЕ 4. Скэн-коды клавиш расширенной клавиатуры компьютеров типа IBM PC.

Клавиша	Код	Shift+ Клавиша	Ctrl+ Клавиша
A	97	65	1
B	98	66	2
C	99	66	3
D	100	68	4
E	101	69	5
F	102	70	6
G	103	71	7
H	104	72	8
I	105	73	9
J	106	74	10
K	107	75	11
L	108	76	12
M	109	77	13
N	110	78	14
O	111	79	15

Клавиша	Код	Shift+ Клавиша	Ctrl+ Клавиша
P	112	80	16
Q	113	81	17
R	114	82	18
S	115	83	19
T	116	84	20
U	117	85	21
V	118	86	22
W	119	87	23
X	120	88	24
Y	121	89	25
Z	122	90	26
1	49	33	-
2	50	64	0
3	51	35	-
4	52	36	-
5	53	37	-
6	54	94	30
7	55	38	-
8	56	42	-
9	57	40	-
0	48	41	-
-	45	95	31
=	61	43	-
[	91	123	27
]	93	125	29
\	124	28	0;43
;	59	58	-
'	39	34	-
,	44	60	0;51
.	46	62	-
/	47	63	-

Клавиша	Код	Shift+ Клавиша	Ctrl+ Клавиша
`	96	126	-
ENTER (доп.клав.)	13	-	10
/ (доп.клав.)	47	47	0;142
* (доп.клав.)	42	0;144	0;78
- (доп.клав.)	45	45	0;149
+ (доп.клав.)	43	43	0;150
5 (доп.клав.)	0;76	53	0;143
F1	0;59	0;84	0;94
F2	0;60	0;85	0;95
F3	0;61	0;86	0;96
F4	0;62	0;87	0;97
F5	0;63	0;88	0;98
F6	0;64	0;89	0;99
F7	0;65	0;90	0;100
F8	0;66	0;91	0;101
F9	0;67	0;92	0;102
F10	0;68	0;93	0;103
F11	0;133	0;135	0;137
F12	0;134	0;136	0;138
HOME (доп.клав.)	0;71	55	0;119
UP (доп.клав.)	0;72	56	0;141
PAGE UP (доп.клав.)	0;73	57	0;132
RIGHT (доп.клав.)	0;75	52	0;115
LEFT (доп.клав.)	0;77	54	0;116
END (доп.клав.)	0;79	49	0;117
DOWN (доп.клав.)	0;80	50	0;145
PAGE DOWN (доп.клав.)	0;81	51	0;118
INSERT (доп.клав.)	0;82	48	0;146
DELETE (доп.клав.)	0;83	46	0;147
HOME	224;71	224;71	224;119
UP	224;72	224;72	224;141

Клавиша	Код	Shift+ Клавиша	Ctrl+ Клавиша
PAGE UP	224;73	224;73	224;132
LEFT	224;75	224;75	224;115
DOWN	224;77	224;77	224;116
END	224;79	224;79	224;117
RIGHT	224;80	224;80	224;145
PAGE DOWN	224;81	224;81	224;118
INSERT	224;82	224;82	224:146
DELETE	224;83	224;83	224;147
PRINT SCREEN	-	-	0;114
PAUSE/BREAK	-	-	0;0
BACKSPACE	8	8	127
ENTER	13	-	10
TAB	9	0;15	0;148
0	0;3	-	-

Скэн - коды клавиш приведены в десятичной форме.

ПРИЛОЖЕНИЕ 5. Назначение управляющих клавиш DOS и клавиш редактирования командной строки.

Enter	указание DOS на необходимость выполнения команды, помещенной в командную строку.
Ctrl-C (Ctrl-Break)	прекращение выполнения пользователем команды DOS или программы.
Ctrl-S (Pause)	приостановка выполнения программы (команды DOS). Повторное нажатие клавиш возобновляет выполнение команды (программы).
PrintScreen (Shift-PrintScreen)	печать копии текущего экрана на принтере.
Ctrl-PrintScreen	параллельная печать всей информации, выводимой на дисплей, на принтер.
Ctrl-Z	оформление конца текстового файла.

F1	копирование из буфера одного символа.
F2	копирование из буфера всех символов, исключая символ, введенный за нажатием клавиши F2.
F3	копирование всего содержимого буфера.
F4	удаление из буфера всех символов, исключая символ, введенный после нажатия клавиши F4.
F5	помещение текущей командной строки в буфер для хранения, но без ее выполнения.
F6	ввод в текстовый файл символа конца файла - EOF. Аналогично нажатию клавиш Ctrl-Z.
Esc	удаление текущей командной строки без ее выполнения, при этом содержимое буфера не меняется.
Del	удаление одного символа в командной строке.

ПРИЛОЖЕНИЕ 6. Объемы, занимаемые различными частями DOS при их загрузке в оперативную память компьютера.

Директивы файла CONFIG.SYS		
Директива	Объем памяти (байт)	Примечания
BREAK	0	
BUFFERS	532	на каждый буфер ввода -вывода
	512	на каждый буфер предввода
COUNTRY	0	
DEVICE	0	+ размер драйвера устройства
DEVICEHIGH	0	+ размер драйвера устройства
DRIVPARM	0	
DOS	0	
FCBS	48	на каждый FCB
FILES	64	на каждый раздел
INSTALL	0	+ размер программы
LASTDRIVE	96	на каждый логический диск
REM	0	
SHELL	0	+ размер командного процессора
STACKS	704	+ размер стеков
SWITCHES	0	

Загружаемые драйверы устройств		
Драйвер	Объем памяти (байт)	Примечания
ANSI.SYS	4208	
DBLSPACE.SYS	339	
DISPLAY.SYS	18064	CON=(EGA,437,1)
DRIVER.SYS	240	
EGA.SYS	3296	
EMM386.EXE	3120	
HIMEM.SYS	1152	
INTERLNK.EXE	9744	
RAMDRIVE.SYS	1200	+ размер электронного (RAM) диска
SETVER.EXE	592	
SMARTDRV.EXE	2496	
POWER.EXE	4672	

Утилиты DOS		
Программа	Объем памяти (байт)	Примечания
APPEND	9040	
DOSKEY	4144	+ размер программного буфера
FASTOPEN	3696	C:=10
	+ 64	при каждом дополнительном файле
GRAPHICS	5936	
KEYB	6224	
MODE	480	все резидентные функции утилиты
NLSFUNC	2784	
PRINT	5792	/B:512 /Q:10 - по умолчанию
	+ 64	при каждом дополнительном /Q:1
SHARE	6208	/F:2048 /L:10 - по умолчанию
	+ 16	при каждом дополнительном /L:1
UNDELETE	13616	
VSAFE	6864	

ПРИЛОЖЕНИЕ 7. Утилиты для Windows.

В состав MS-DOS версии 6.0 входит несколько утилит, представляющих версии утилит DOS для среды Microsoft Windows. К этим программам относятся следующие:

"MWAV" - версия антивируса Microsoft Anti-Virus (MSAV) для Windows;

"MWAVTSR" - версия резидентного антивируса VSAFE под Windows;

"MWBACKUP" - версия утилиты MSBACKUP для Windows;

"MWUNDEL" - версия утилиты UNDELETE в среде Windows;

"SmartMon" - программа управления драйвером SMARTDRV.EXE в среде Windows.

Все эти программы могут быть включены в состав Windows при инсталляции оболочки. Если же вы будете устанавливать MS-DOS 6.0 после того, как Windows уже имеется на компьютере, либо вы не включили в состав Windows утилиты DOS при ее инсталляции, необходимо подключить Windows-приложения стандартными для Windows способами.

Вопросы работы с Windows широко обсуждаются во многих изданиях, например, в недавно вышедшей книге Стефана Фойца "Windows 3.1".

Список литературы

1. Р. Джордейн. "Справочник программиста персональных компьютеров IBM PC, XT и AT." Москва, "Финансы и статистика". 1992 г.

2. П. Нортон, Р.Джордейн. "Работа с жестким диском IBM PC". Москва,"Мир",1992 г.

3. В. Л. Григорьев. "Видеосистемы ПК фирмы IBM". Москва, "Радио и связь". 1993 г.

4. А. Я. Савельев, Б. А. Сазонов, С. Э. Лукьянов. "Персональный компьютер для всех". Выпуски 1-4. Москва, "Высшая школа", 1991 г.

5. К. Г. Финогенов. "MS-DOS 5.0". Части 1-2. Москва, МП "Малип", 1992 г.

6. В. М. Брябрин. "Программное обеспечение персональных ЭВМ". Москва, 1991 г.

7. "PC Magazine. Russian Edition". 1992 - 1993 гг.

8. "Мир ПК". 1991 - 1993 гг.

9. "MS-DOS 6.0 User Guide". Microsoft Corp., 1993 г.

10. "MS-DOS 5.0 User Guide". Microsoft Corp., 1991 г.

СОДЕРЖАНИЕ

Введение . . . 3

Организация IBM PC и требования MS-DOS 6.0 к ПЭВМ . . . 5

Состав DOS и ее основные модули . . . 8
- Базовая система ввода-вывода (BIOS) . . . 9
- Блок начальной загрузки (Boot Record) . . . 10
- Модуль расширения базовой системы ввода-вывода (IO.SYS) . . . 10
- Модуль обработки прерываний (MSDOS.SYS) . . . 11
- Командный процессор (COMMAND.COM) . . . 12
- Модуль сжатия данных (DBLSPACE.BIN) . . . 14
- Утилиты DOS . . . 14
- Загружаемые драйверы устройств . . . 14
- Системное окружение DOS . . . 15
- Файл конфигаурации системы CONFIG.SYS . . . 17
- Файл автоматического запуска AUTOEXEC.BAT . . . 20
- Ввод информации с клавиатуры в MS-DOS . . . 21

Организация хранения данных в MS-DOS . . . 26
- Файлы . . . 26
- Файловая структура диска . . . 29
  - Таблица размещения файлов (FAT) . . . 32
  - Каталоги . . . 34
  - Командные файлы . . . 43
  - Перенаправление ввода-вывода . . . 45
  - Фильтры и конвейер . . . 46
- Внутренние команды пакетных файлов . . . 47

Команды пакетных файлов . . . 49
- @ . . . 49
- CALL . . . 49
- (!) CHOICE . . . 50
- ECHO . . . 52
- FOR . . . 53
- GOTO . . . 54
- IF . . . 54
- PAUSE . . . 56
- REM . . . 57
- SHIFT . . . 58

Директивы файла CONFIG.SYS . . . 59
- BREAK . . . 59
- BUFFERS . . . 59
- COUNTRY . . . 60
- DEVICE . . . 61
- DEVICEHIGH . . . 61
- DOS . . . 63
- DRIVPARM . . . 63
- FCBS . . . 65
- FILES . . . 65
- INSTALL . . . 66
- LASTDRIVE . . . 66
- REM . . . 66
- SHELL . . . 67

STACKS . . . 68
SWITCHES . . . 68
INCLUDE . . . 70
MENUCOLOR . . . 71
MENUDEFAULT . . . 72
MENUITEM . . . 73
SUBMENU . . . 73
NUMLOCK . . . 75

Загружаемые драйверы устройств . . . 76
ANSI.SYS . . . 76
CHKSTATE.SYS . . . 77
DBLSPACE.SYS . . . 77
DISPLAY.SYS . . . 78
DRIVER.SYS . . . 79
EGA.SYS . . . 81
EMM386 . . . 81
HIMEM.SYS . . . 86
(!) INTERLNK.EXE . . . 89
(!) POWER . . . 91
RAMDRIVE.SYS . . . 92
(!) SETVER.EXE . . . 93
SMARTDRV.EXE . . . 94

Команды DOS . . . 96
APPEND . . . 96
ATTRIB . . . 98
BREAK . . . 99
CHCP . . . 100
CHDIR (CD) . . . 101
CHKDSK . . . 103
CLS . . . 105
COMMAND . . . 106
COPY . . . 107
CTTY . . . 112
DATE . . . 113
(!) DBLSPACE . . . 114
DEBUG . . . 125
(!) DEFRAG . . . 146
DEL (ERASE) . . . 151
(!) DELTREE . . . 153
DIR . . . 154
DISKCOMP . . . 160
DISKCOPY . . . 162
DOSKEY . . . 165
DOSSHELL . . . 170
(!) EDIT . . . 174
ERASE . . . 179
EXIT . . . 179
EXPAND . . . 180
(!) FASTHELP . . . 181
FASTOPEN . . . 182
FC . . . 184
FDISK . . . 187
FIND . . . 190
FORMAT . . . 192
GRAPHICS . . . 196

HELP . . . 199
(!) INTERLNK . . . 201
(!) INTERSVR . . . 204
KEYB . . . 206
LABEL . . . 209
(!) LOADFIX . . . 211
LOADHIGH (LH) . . . 212
MEM . . . 214
(!) MEMMAKER . . . 218
MKDIR (MD) . . . 221
MODE . . . 221
MORE . . . 231
(!) MOVE . . . 232
(!) MSAV . . . 234
(!) MSBACKUP . . . 239
(!) MSCDEX . . . 248
(!) MSD . . . 250
NLSFUNC . . . 254
PATH . . . 256
(!) POWER . . . 257
PRINT . . . 258
PROMPT . . . 261
RENAME (REN) . . . 263
REPLACE . . . 264
(!) RESTORE . . . 266
RMDIR (RD) . . . 269
SET . . . 270
SETVER . . . 272
SHARE . . . 274
(!) SMARTDRV . . . 276
SORT . . . 280
SUBST . . . 282
SYS . . . 283
TIME . . . 284
TREE . . . 286
TYPE . . . 287
UNDELETE . . . 288
UNFORMAT . . . 295
VER . . . 296
VERIFY . . . 297
VOL . . . 297
(!) VSAFE . . . 298
XCOPY . . . 300

Приложения . . . 304
1. Сводный список команд MS-DOS по версиям. . . . 304
2. Список внутренних команд DOS. . . . 307
3. Esc-последовательности драйвера ANSI.SYS. . . . 307
4. СКЭН-коды клавиш расширенной клавиатуры компьютеров IBM PC . . 309
5. Назначение управляющих клавиш DOS и клавиш редактирования командной строки. . . . 312
6. Объемы, занимаемые различными частями DOS при их загрузке в оперативную память компьютера. . . . 313
7. Утилиты для Windows . . . 315

СПИСОК ЛИТЕРАТУРЫ . . . 315

Книга представляет собой справочное пособие по основам построения и функционирования последней версии операционной системы MS-DOS 6.0, которая используется на персональных компьютерах семейства IBM PC и совместимых с ними.

В начале книги содержатся общие сведения об основных модулях MS-DOS, их взаимодействии, рассмотрены вопросы организации файловой структуры и хранения данных на магнитных дисках.

Далее книга содержит сведения о файлах AUTOEXEC.BAT и CONFIG.SYS в MS-DOS 6.0, файлах пакетной обработки, командах пакетных файлов и директивах файла CONFIG.SYS.

Основную часть книги занимает описание команд MS-DOS 6.0. Приведены краткие описания всех команд MS-DOS 6.0, форматы команд, рассмотрены примеры их использования и особенности работы с командами операционной системы.

Для широкого круга пользователей персональных компьютеров IBM PC, PC/XT, PC/AT и совместимых с ними.

Ваши отзывы о данном издании, предложения о сотрудничестве просим присылать по адресу:

123362, Москва, а/я 26

MS-DOS 6.0

Справочное руководство для пользователей компьютеров IBM PC

Ответственный за выпуск: *Васятко В.Н.*

Технический редактор: *Шелепова Е.Н.*
Панушкина О.В.
Константинова С.С.

Компьютерная верстка: *Тарасов С.Ю.*
Мельниченко А.Н.

Издательство «ВА Принт».
Лицензия на издательскую деятельность ЛР 061721 от 23.10.92.
Издательский код 6Б8(03).
123362 Москва, а/я 26

Подписано в печать 8.12.93. Формат 60×88 1/16. Заказ 6805. Тираж 20000 экз. Государственное ордена Октябрьской Революции, ордена Трудового Красного Знамени Московское предприятие «Первая Образцовая типография» Министерства печати и информации Российской Федерации.
113114, Москва, М-114, Шлюзовая набережная, 10.