

# Подсистема ввода-вывода. Файловые системы

---

1. Принципы функционирования аппаратуры ввода-вывода
2. Обработчики прерываний и драйверы устройств
3. Понятие файла и каталога
4. Задачи ОС по управлению файлами
5. Структура файловой системы и виды файловых систем
6. Управление дисковыми ресурсами

# 1. Принципы функционирования аппаратуры ввода-вывода

---

## **Физическая организация устройств**

Устройства ввода-вывода делятся на два типа:

- *блок-ориентированные* устройства
- *байт-ориентированные* устройства

Блок-ориентированные устройства хранят информацию в блоках фиксированного размера, каждый из которых имеет свой собственный адрес (диск).

Байт-ориентированные устройства не адресуемы и не позволяют производить операцию поиска, они генерируют или потребляют последовательность байтов (принтер).

---

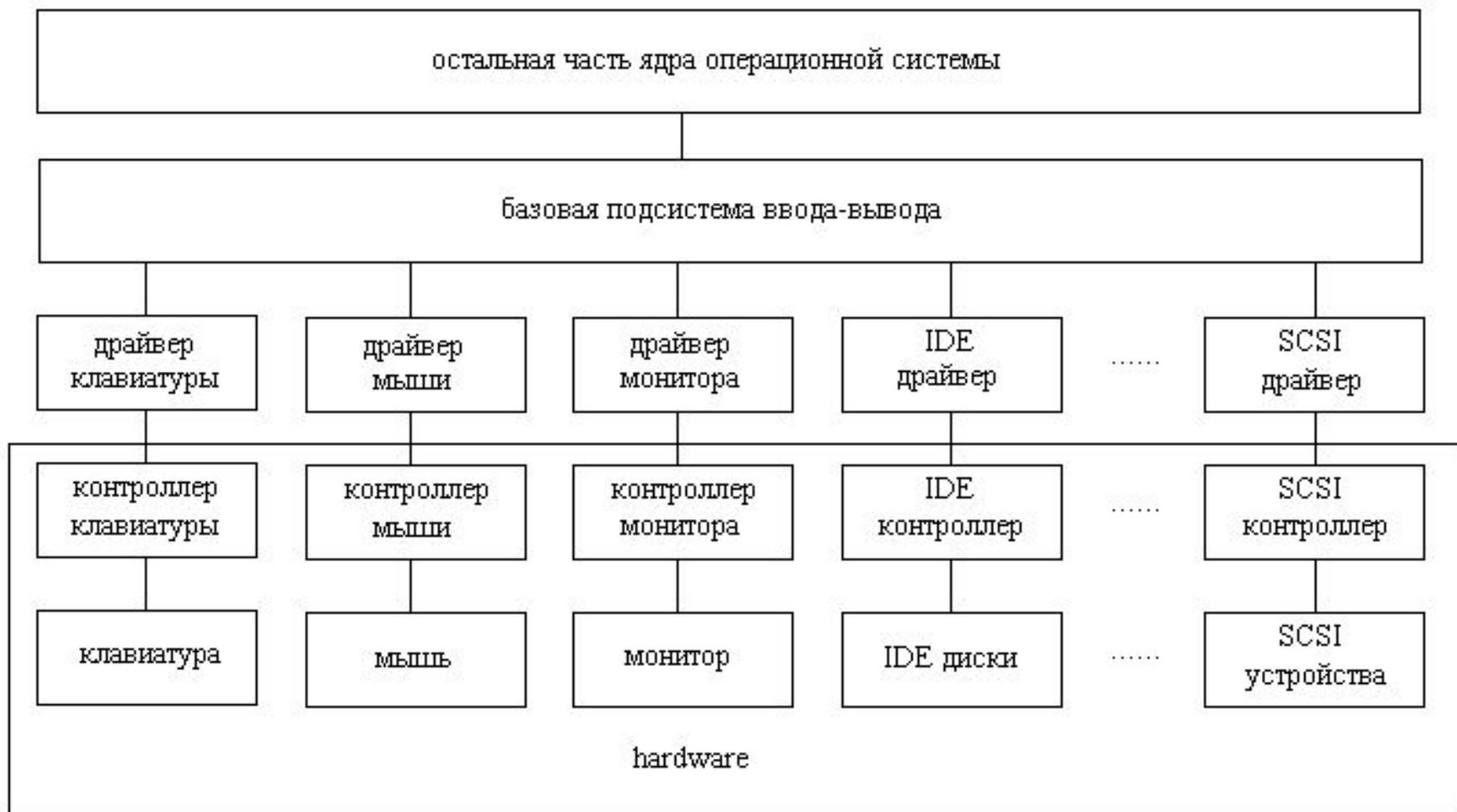
Внешнее устройство обычно состоит из электронного механического и компонентов.

Электронный компонент называется **контроллером** устройства или адаптером.

Механический компонент представляет собственно устройство.

Операционная система взаимодействует не с устройством, а с контроллером.

Каждый контроллер имеет несколько регистров. ОС выполняет ввод-вывод, записывая команды в регистры контроллера.



# Организация программного обеспечения ввода-вывода

---

Идея организации программного обеспечения ввода-вывода состоит в разбиении его на несколько уровней.

Нижние уровни обеспечивают экранирование особенностей аппаратуры от верхних.

Основная задача **независимого от устройств** программного обеспечения заключается в выполнении функций ввода/вывода, общих для всех устройств, и предоставлении единообразного интерфейса для программ уровня пользователя.

---

В составе программного обеспечения ввода-вывода выделяют четыре слоя:

- Обработка прерываний
- Драйверы устройств
- Независимый от устройств слой операционной системы
- Пользовательский слой программного обеспечения

# Многоуровневая организация подсистемы ввода-вывода



## 2. Обработчики прерываний и драйверы устройств

---

Наилучший способ состоит в разрешении процессу, инициировавшему операцию ввода-вывода, блокировать себя до завершения операции и наступления прерывания.

Эффект от прерывания будет состоять в том, что ранее заблокированный процесс теперь продолжит свое выполнение.

**Драйвер** - программа, с помощью которой другие программы (в т.ч. ОС) получают доступ к аппаратному обеспечению некоторого устройства.

---

Операционная система управляет некоторым «виртуальным устройством», которое понимает стандартный набор команд.

Драйвер переводит эти команды в команды, которые понимает непосредственно устройство. Т.е. весь зависимый от устройства код помещается в драйвер устройства.

Каждый драйвер управляет устройствами одного типа или, может быть, одного класса.

## **Независимый от устройств слой операционной системы**

---

Основная задача независимого от устройств программного обеспечения заключается в выполнении функций ввода/вывода, общих для всех устройств, и предоставлении единообразного интерфейса для программ уровня пользователя.

Большая часть программного обеспечения ввода-вывода является независимой от устройств.

Точная граница между драйверами и независимыми от устройств программами определяется ОС.

Типичными функциями для независимого от устройств слоя являются:

---

- обеспечение общего интерфейса к драйверам устройств,
- именованное устройств,
- защита устройств,
- обеспечение независимого размера блока,
- буферизация,
- распределение памяти на блок-ориентированных устройствах,
- распределение и освобождение выделенных устройств,
- уведомление об ошибках.

## 3. Понятие файла и каталога

---

*Файл* – логически связанная совокупность данных или программа, записанная в виде поименованной области на устройстве хранения.

*Файловая система* (ФС) является составной частью любой операционной системы и отвечает за организацию хранения и доступа к информации на каких-либо носителях.

---

В широком смысле понятие "файловая система" включает:

- совокупность всех файлов на диске,
- наборы служебных структур данных, используемых для управления файлами (каталоги, таблицы распределения свободного и занятого пространства на диске, атрибуты файлов и т.д.)
- набор функций по управлению файлами (создание, удаление, чтение, запись, установка атрибутов и уровней доступа и т.д.)

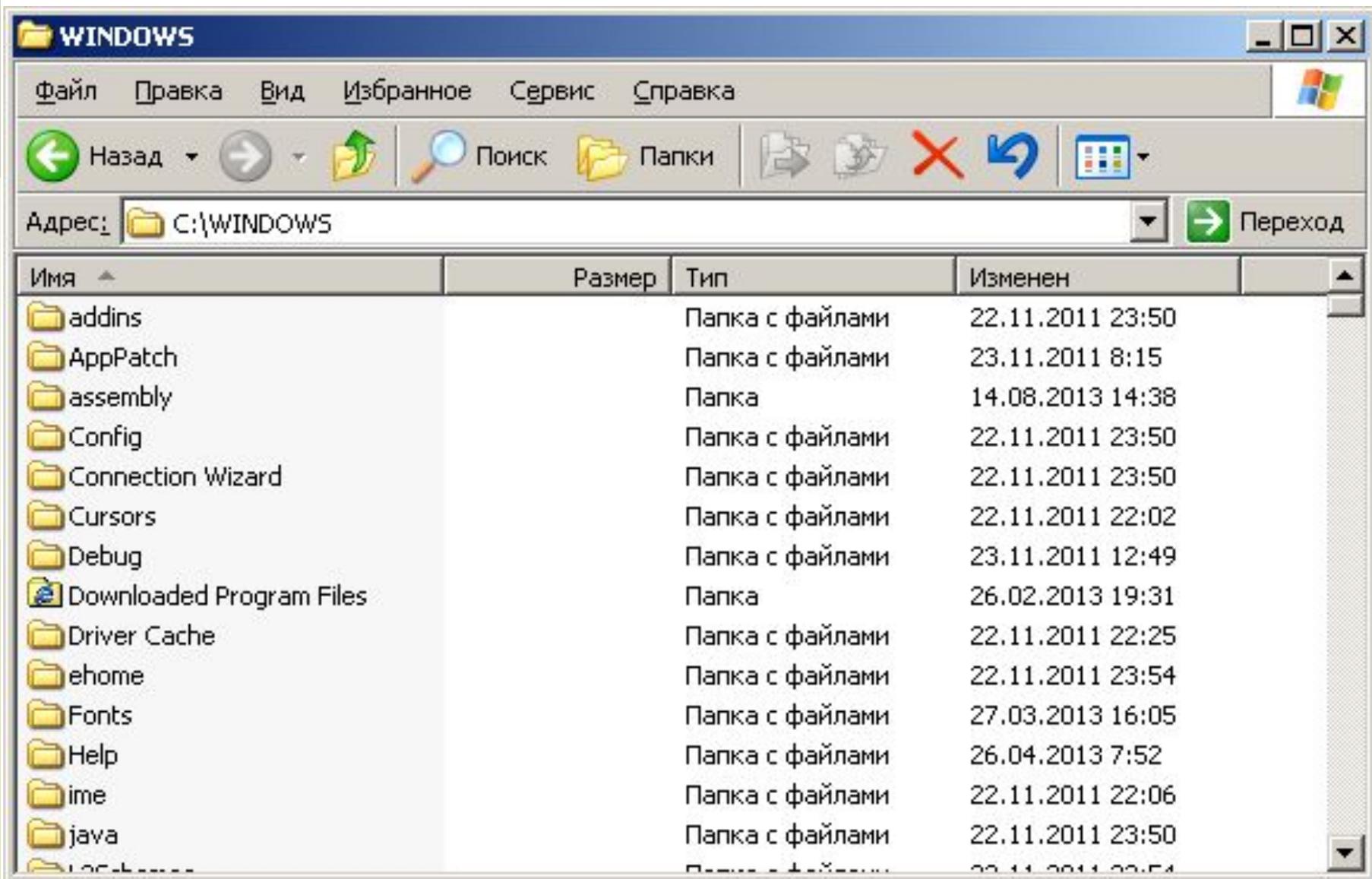
## Базовые функции файловой системы:

- обеспечение оперативной работы с файлами: создание, модификация, удаление;
- обеспечение пользователям возможности контролируемого доступа к некоторому подмножеству файлов в системе;
- обеспечение пользователям возможности устанавливать тип доступа к их файлам;
- обеспечение возможности пользователям изменять структуру своих файлов в соответствии с текущей задачей;
- обеспечение возможности пересылки данных между файлами;
- обеспечение возможности резервного копирования, архивации, компрессии файлов;
- обеспечение возможности копирования и восстановления файлов в случае их повреждения;
- обеспечение доступа к файлам по символическим именам.

Каталог (*directory* — справочник, указатель) — объект в файловой системе (файл особого вида), группирующий файлы и подкаталоги и упрощающий организацию хранения файлов на диске.

Корневой каталог находится на самом верхнем уровне иерархии всех каталогов и не является подкаталогом ни одного другого каталога.

Термин папка (*folder*) был введён для представления объектов файловой системы (каталогов) в графическом пользовательском интерфейсе.



## 4. Задачи ОС по управлению файлами

---

- создание, удаление, переименование и другие операции с файлами;
- работа с не дисковыми периферийными устройствами как с файлами;
- обмен данными между файлами, между устройствами, между файлом и устройством (и наоборот);
- работа с файлами с помощью обращений к программным модулям системы управления файлами;
- защита файлов от несанкционированного доступа.

## 5. Структура файловой системы и виды файловых систем

---

Файловая система имеет два уровня организации:

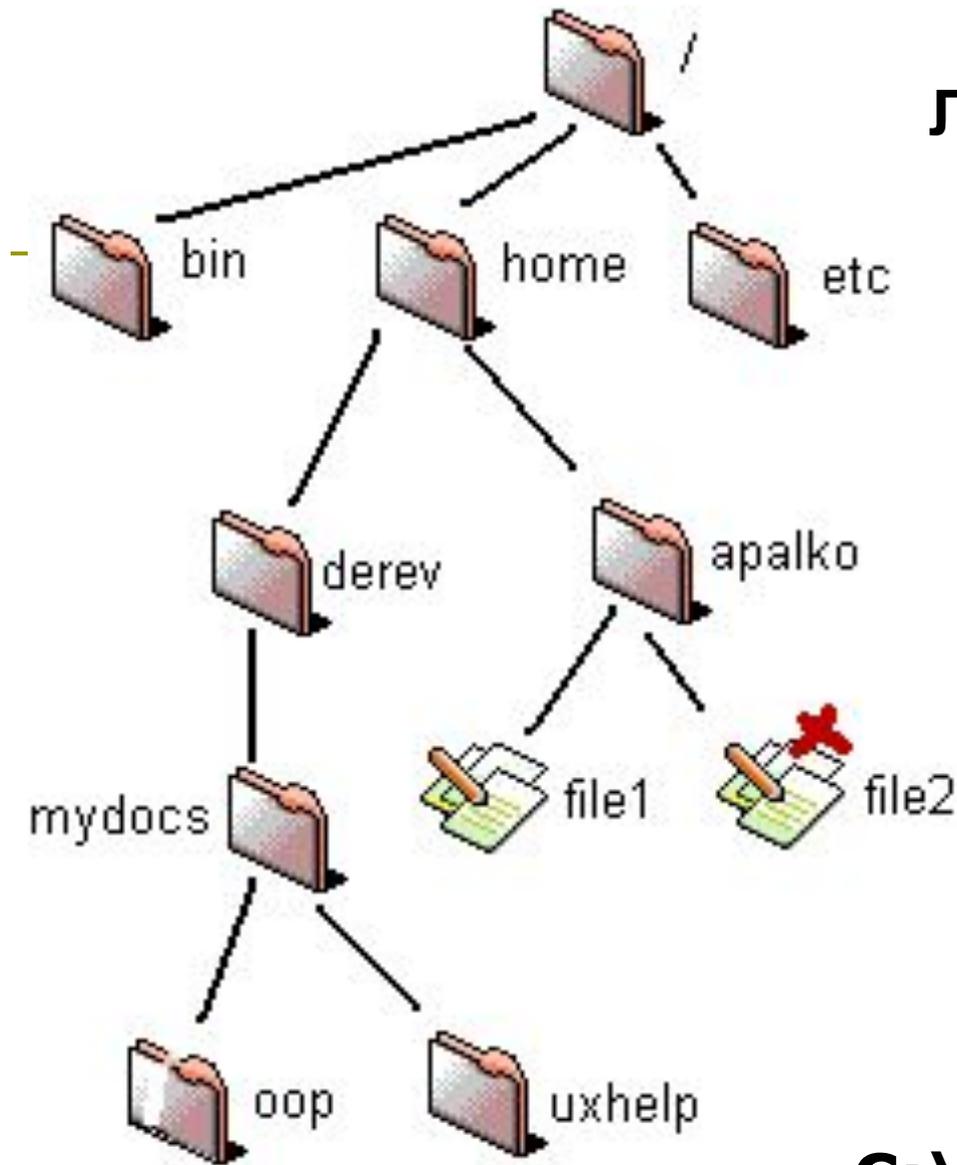
- ▣ *логический*
- ▣ *физический.*

На **логическом уровне** описывается относительное местоположение файлов в компьютере.

Файловая система имеет иерархическую структуру хранения файлов, в которой уровни создаются за счет каталогов.

**Физический уровень** отвечает за управление дисковым пространством и размещение файлов в физических адресах диска.

## Логический уровень



### *Путем к файлу*

называется цепочка символов, начиная с имени дисководов, корневого каталога и последующих подкаталогов вплоть до каталога, содержащего необходимый файл.

**C:\home\apalko\file1**

# Характеристики некоторых файловых систем

Название	Максимальная длина имён файлов	Максимальный размер файла	Максимальный размер тома	Фирма-разработчик	Операционная система
RT-11	6+3 символа в коде <b>RADIX50</b>	33 554 432 байт (65536 * 512)	33 554 432 байт	DEC	RT-11
FAT12	8+3 символов (255 байт для VFAT)	32MiB	1MiB — 32MiB	Microsoft	Microsoft Disk BASIC
FAT16	8+3 символов (255 байт для VFAT)	2GiB	16MiB — 2GiB	Microsoft	MS-DOS 2.0
FAT32	255 байт	4GiB	512MiB — 8TiB	Microsoft	Windows 95b
HPFS	255 байт	4GiB	2TiB	IBM & Microsoft	OS/2
NTFS	255 символов	16 EiB	16 EiB	Microsoft, Gary Kimura, Tom Miller	Windows NT
HFS+	255 символов	8EiB	8EiB	Apple	Mac OS
FFS	255 байт	4GiB	256TiB	Kirk McKusick	4.2BSD
ext2	255 байт	16GiB — 2TiB	2TiB — 32TiB	Rémy Card	Linux
ext3	255 байт	16GiB — 2TiB	2TiB — 32TiB	Stephen Tweedie	Linux
ext4	255 байт	16GiB — 16TiB	1 EiB	Andrew Morton	Linux
JFS	255 байт	8EiB	512TiB — 4PiB	IBM	AIX[3]
JFS2	255 байт	4PiB	32PiB	IBM	OS/2 WSeB
NSS	256 символов	8TiB	8TiB	Novell	NetWare 5
NWFS	80 байт	4GiB	1TiB	Novell	NetWare 286
UDF	255 байт	16EiB	?	ISO/ECMA/OSTA	-
ZFS	255 байт	16EiB	16EiB	Sun Microsystems	Solaris
exFAT	Неизвестно	16EiB	64 ZiB	Microsoft	Windows Vista SP1+

## 6. Управление дисковыми ресурсами

---

Все пространство магнитного диска разбито на *дорожки*, которые в свою очередь разделены на *сектора*.

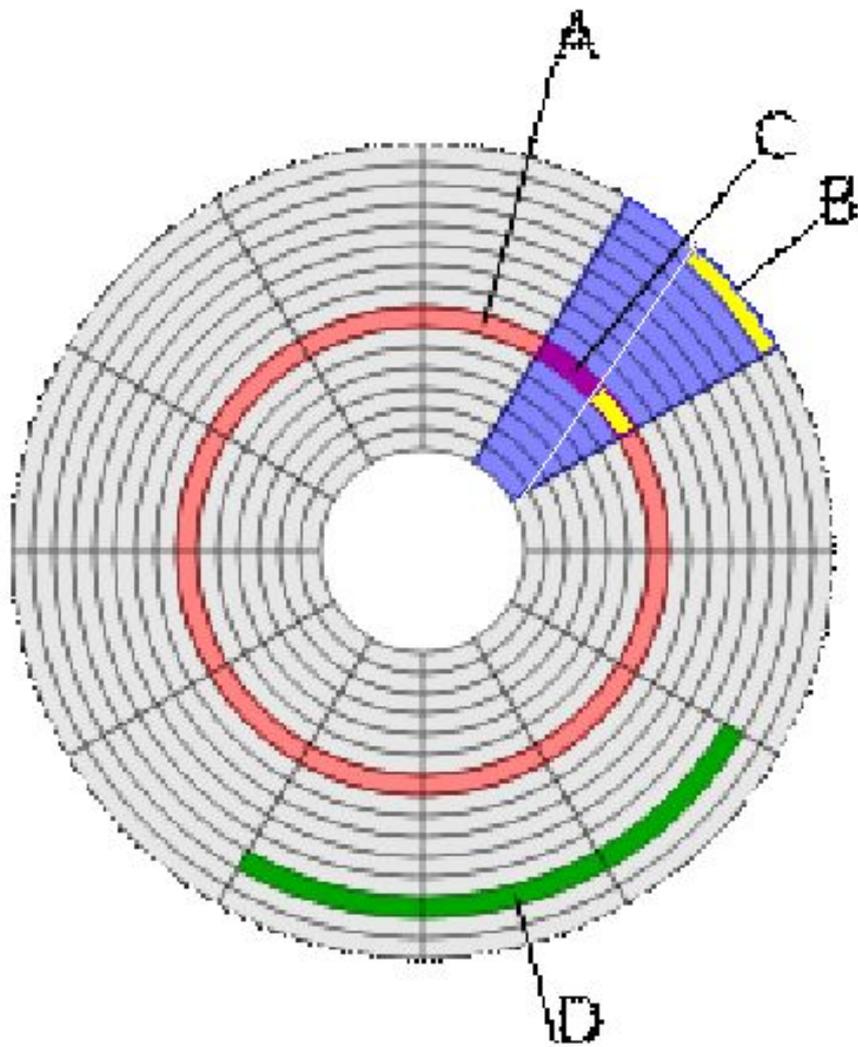
Для создания дорожек и секторов, а так же служебных данных файловой системы используется специальная процедура – *форматирование*, которая выполняется, обычно, с помощью программы FORMAT.

---

Процесс форматирования является сложной процедурой, состоящей из двух отдельных операций, называемых форматированием *НИЗКОГО* и *ВЫСОКОГО* уровней.

Низкий уровень форматирования (физический) состоит в нанесении на диск электронных меток для указания физических мест дорожек и секторов.

Затем на *высоком уровне форматирования*, называемом также *логическим*, происходит выделение служебных областей на диске в соответствии с видом файловой системы.



Структура диска:

(A) дорожка

(B) межсекторный промежуток

(C) сектор дорожки

(D) кластер

---

Размер сектора (точнее – емкость поля данных) устанавливается контроллером или драйвером.

**Кластер** (*cluster*) — в некоторых видах файловых систем логическая единица хранения данных, объединяющая группу секторов.

С точки зрения операционной системы, весь диск представляет собой набор кластеров (размером 512 байт или кратного 512).

---

Жесткий диск может быть разбит на несколько *разделов (partition)*, которые затем могут использоваться либо одной, либо различными ОС. В каждом разделе может быть организована своя файловая система. Поддержка той или иной файловой системы осуществляется драйверами. Разделы диска могут быть *primary (первичный)* и *extended (расширенный, дополнительный)*. Максимальное количество разделов на диске – 4 или все первичные или 3 первичных и **только 1** расширенный, содержащий до 4 логических разделов.

---

На диске **обязательно** должен быть по крайней мере **один** primary-раздел.

Если primary-разделов несколько, то **только один** из них может быть **активным** (он используется для загрузки ОС).

Дополнительный раздел может быть **только один**, но его структура позволяет создавать внутри него большое число **логических дисков**.

Partition	Type	Size MB	Used MB	Unused MB	Status	Pri/Log
<b>Disk 1</b>						
SYS (C:)	■ NTFS	11 248.6	10 312.1	936.5	Active	Primary
WRK1 (D:)	■ NTFS	10 707.4	7 853.8	2 853.6	None	Primary
WRK2 (E:)	■ NTFS	54 360.6	45 010.6	9 350.0	None	Primary

В первом физическом секторе жесткого диска располагается **головная загрузочная запись (MBR)** и **таблица разделов диска (*partition table*)**.

Каждая запись в таблице разделов содержит начальную позицию и размер раздела на жестком диске, а также информацию о том, первый сектор какого раздела содержит загрузочную запись.

BIOS считывает и загружает в память MBR и далее первый физический сектор активного раздела, называемый **загрузочным сектором (*Boot Sector*)**.

# Служебные области диска в файловой системе FAT16

---

Загрузочный сектор	FAT	FAT (копия)	Корневой каталог	Область данных
--------------------	-----	-------------	------------------	----------------

## Алгоритм считывания файлов в FAT

1	2	3	4	5	6	7
				<b>10</b>		
8	9	10	11	12	13	14
		<b>14</b>				<b>19</b>
15	16	17	18	19	20	21
				<b>27</b>		
22	23	24	25	26	27	28
					<b>EOF</b>	
29	30	31	32	33	34	...

цепочка кластеров, принадлежащих данному файлу – **5,10,14,19,27**

**Каждому файлу и подкаталогу в FAT выделяется  
32-байтный элемент каталога :**

<b>Содержание</b>	<b>Размер (байт)</b>
<b>Имя файла</b>	<b>8</b>
<b>Расширение</b>	<b>3</b>
<b>Байт атрибутов</b>	<b>1</b>
<b>Зарезервировано</b>	<b>10</b>
<b>Время</b>	<b>2</b>
<b>Дата</b>	<b>2</b>
<b>Номер начального кластера с данными</b>	<b>2</b>
<b>Размер файла</b>	<b>4</b>

# Состав дискового раздела NTFS

\$MFT	сам MFT (Master File Table)
\$MFTmirr	копия первых 16 записей MFT, размещенная посередине диска
\$LogFile	файл поддержки журналирования
\$Volume	служебная информация - метка тома, версия файловой системы, т.д.
\$AttrDef	список стандартных атрибутов файлов на томе
\$.	корневой каталог
\$Bitmap	карта свободного места тома

\$Boot	загрузочный сектор (если раздел загрузочный)
\$Quota	файл, в котором записаны права пользователей на использование дискового пространства (начал работать лишь в NT5)
\$Upcase	файл - таблица соответствия заглавных и прописных букв в имен файлов на текущем томе. Нужен в основном потому, что в NTFS имена файлов записываются в Unicode.

МФТ поделена на записи фиксированного размера  
(обычно 1 Кбайт)

---

Каждая запись соответствует одному файлу.

Первые 16 файлов носят служебный характер и недоступны операционной системе - они называются метафайлами, причем самый первый метафайл - сам МФТ.

Эти первые 16 элементов МФТ - единственная часть диска, имеющая фиксированное положение.

# Сравнение файловых систем

<b>Характеристика</b>	<b>FAT</b>	<b>VFAT</b>	<b>FAT32</b>	<b>NTFS</b>
<b>Максимальный размер раздела</b>	<b>2 Гб</b>	<b>4 Гб</b>	<b>4 Тб</b>	<b>16 Эб</b>
<b>Максимальный размер файла</b>	<b>2 Гб</b>	<b>4 Гб</b>	<b>4Тб</b>	<b>16 Эб</b>
<b>Максимальное количество файлов в корневом каталоге</b>	<b>512</b>	<b>512</b>	<b>Неогр.</b>	<b>Неогр.</b>
<b>Максимальное количество файлов в некорневом каталоге</b>	<b>65535</b>	<b>Неогр.</b>	<b>Неогр.</b>	<b>Неогр.</b>
<b>Безопасность на уровне файлов</b>	<b>Нет</b>	<b>Нет</b>	<b>Нет</b>	<b>Да</b>
<b>Поддержка длинных имен файлов</b>	<b>Нет</b>	<b>Да</b>	<b>Да</b>	<b>Да</b>
<b>Самовосстановление</b>	<b>Нет</b>	<b>Нет</b>	<b>Да</b>	<b>Да</b>
<b>Ведение журналов транзакций</b>	<b>Нет</b>	<b>Нет</b>	<b>Нет</b>	<b>Да</b>
<b>Сжатие на уровне файлов</b>	<b>Нет</b>	<b>Нет</b>	<b>Нет</b>	<b>Да</b>