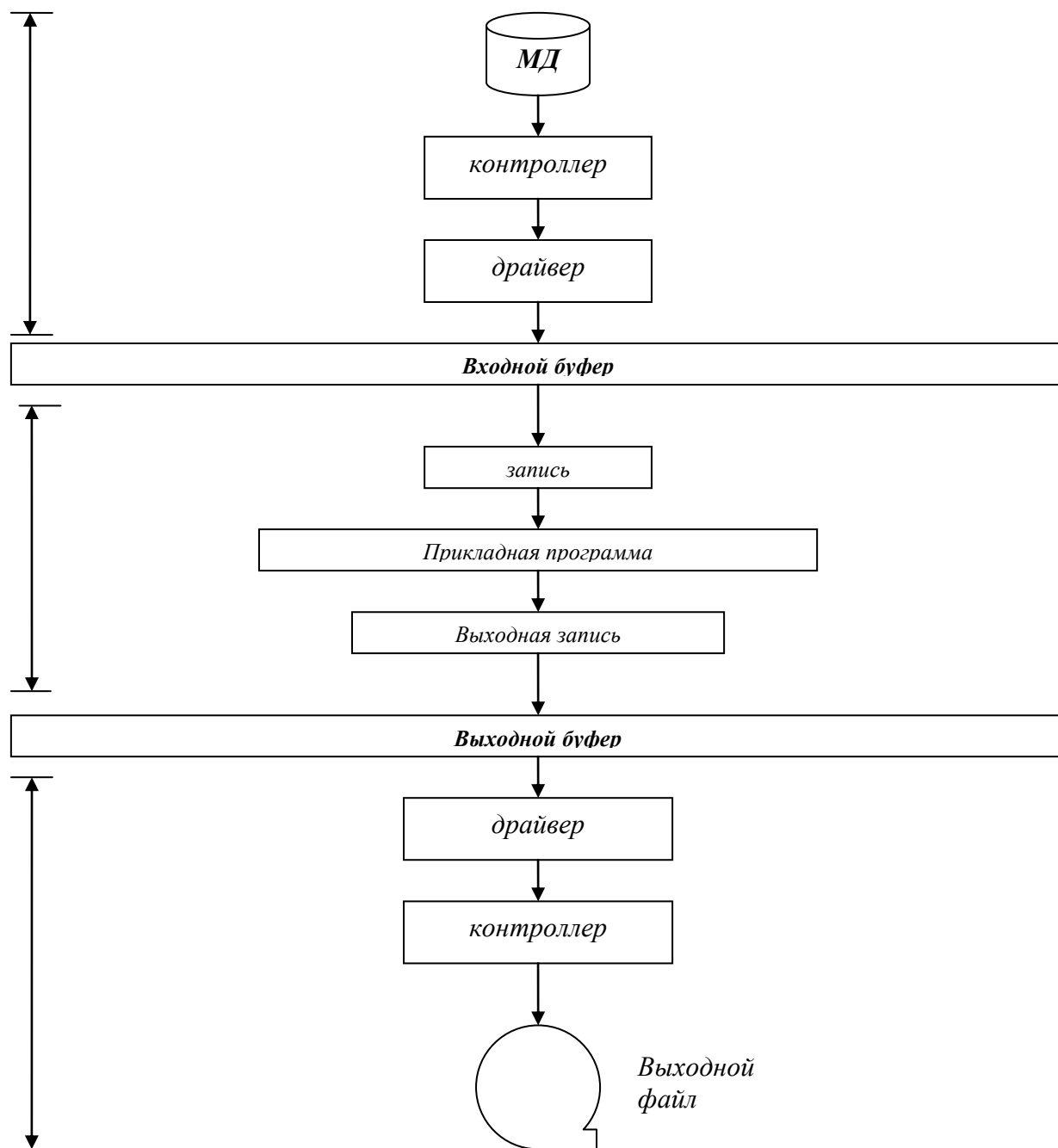


2.4 Машинно-независимые свойства ОС.

2.4.1 Организация данных. Методы доступа.

Схема прохождения данных при обработке в ЭВМ.



Данные – любая информация, которая должна быть обработана ПК.

Поле – законченная по смыслу порция данных.

Запись – набор полей, содержащих множество данных, необходимых для выполнения 1 цикла программы.

Если поле – графа таблицы, то запись – строка:

ФИО	Год рождения	Адрес	Телефон
Босько И.Н.	24.11.1976	Г. Тосно	25983

Файл – набор соотносимых подобных записей, которая оформляется на внешних запоминающих устройствах (поименованная область на диске).

В ОП данные записываются в последовательных ячейках памяти. В зависимости от типа значения данные занимают определенное количество байтов. Все байты в ОП адресуются, поэтому можно обратиться на прямую к любому байту за выборкой необходимых данных. Тип адресации для каждого МП типа свой и указывается в архитектуре.

На ВЗУ данные хранятся в виде файлов и записываются в специально организованные области. Адрес включает в себя указание № цилиндра Т, № поверхности Н, № сектора S [t-h-s]. Информация на МД записывается по секторам и дорожкам, причем записи, принадлежащие одному файлу, могут находиться на разных дорожках.

FDD - Число дорожек составляет 40 или 80; 2 рабочие поверхности.

HDD - Может иметь 305-614 и другое количество цилиндров.

Количество рабочих поверхностей и цилиндров является аппаратной характеристикой дисководов.

Программно управляя контроллером дисководов можно изменить число цилиндров.

Число секторов на дорожке задается программно драйвером.

FDD – 8; 9; 15; 18

HDD – 17; 32 и более

Каждый сектор состоит из поля данных и поля служебной информации, которая ограничивает и идентифицирует его.

MSDOS поддерживает размер сектора 512 б.

BIOS Поддерживает 128; 256; 512; 1024

Число цилиндров, секторов и размер сектора устанавливаются при форматировании диска (формат, пробел, логическое имя диска).

Обмен информации между ОЗУ и МД физически осуществляется только секторами.

При создании файла ОС выделяет для файла дисковое пространство блоками – кластерами.

Кластер – минимальный объем дисковой памяти, выделенный файлу.

Методы доступа к данным – это алгоритм запоминания поиска записи в файле.

Существуют методы:

1. последовательный;
2. индексно-последовательный;
3. прямой.



В зависимости от метода файл должен иметь ту или иную логическую структуру для обеспечения доступа к записи файла.

- (1) данный метод применяем к любой организации файлов, при этом к нужной записи обращаемся, просмотрев все предыдущие;
- (2) все записи в файле связаны между собой в цепочку таким образом, что каждый из них имеет ссылку на номер последующей. Поэтому, обратившись к какой-либо записи можно раскрутить всю цепочку последующих. Недостаток в том, что сложно включить новые записи в цепочку;
- (3) применим только к файлам, имеющим такую логическую структуру, при которой каждая запись строго фиксирована по количеству байтов, причем физически каждая запись имеет свой номер, начиная с 0, поэтому, указав номер интересующей нас записи можно напрямую обратиться к ней, не просматривая другие.

Устройства, поддерживающие *прямой метод доступа*: FDD, HDD

Последовательный доступ: strimer, терминальные и печатающие устройства.

Различают:

- | | | | | | | |
|---|---|------|---|---|---|---|
| 0 | 1 | 2... | 5 | 6 | п | П+1 |
|  | | | |  | | |
| каталог | | | | | | |

файлы

Каталог размещен с 6 блока и содержит совокупность сегментов (сегмент = 2 блока).

- общее количество сегментов в каталоге;
- номер следующего сегмента;
- счетчик занятых сегментов;
- номер блока, начиная с которого начинается файл;

Все дисковое пространство разделяется на кластеры.

FAT(File Allocation Table)

Каталог – это специальный файл, в котором регистрируются другие файлы. Все операции над файлами доступны через прерывания верхнего уровня 2Dh – 3Fh.

Чтобы воспользоваться услугами ФС по выполнению операции над элементами файловой структуры в программе следует выдать соответствующее прерывание (21h), а также предусмотреть загрузку необходимых аргументов в режиме МП. Сведения о файле ОС получает с помощью описателя файла. Получив управление, ФС идентифицирует запрос на выполнение операции (создать – CREATE, открыть – OPEN, закрыть – CLOSED, читать – READ, записать – WRITE), выбирает требуемый для этого драйвер и транслирует запрос в последовательность обращений к драйверу. Драйвер обеспечивает непосредственное управление контроллером МД по выполнению требуемой операции (через порт ввода - вывода), выдавая или принимая последовательность команд, а также анализируя операции их выполнения. После завершения операции драйвер возвращает управление и результаты своей работы ядру файловой системы, которая в свою очередь передает результаты программе, выдавший запрос.

Элементы FAT.

Дескриптор тома	1 доступный кластер	2 кластер	3 кластер	4 кластер
0	1 2	3	4	5

Смысл кодировки элементов в FAT.

0000h – свободен.

0001h – FFEFh – занят (указывается № кластера).

FFF0h – FFF6h – зарезервирован.

FFF7h – дефективный.

FFF8h – FFFFh – конец цепочки кластеров.

FAT 16 – означает, что для кодировки каждого элемента используется 16 бит.

DT	1	2	3	4
		+		
		0003 h	0004 h	→0004 h
5	6	7	8	9
10	11	12	13	14
+				
FFFF h				
15	16	17

+ - свободен
1 кл = 512 б
2048б – длина файла в байтах

При создании файла в элемент каталога заносится: - file.txt – имя файла.

- атрибут – архивный.

- 19 октября 2004 – дата.

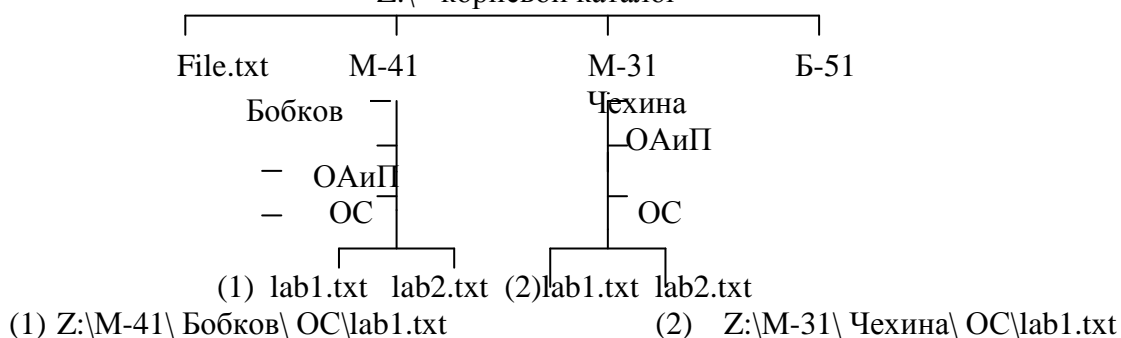
- 0002h – первый кластер, с которого начинается цепочка.

Таким образом, любой элемент каталога ФС может быть найден по указанию его имени, так как с именем связан номер начального кластера, с которого начинается файл, а затем ФС обращается к FAT, где указана вся цепочка кластеров, принадлежащих данному файлу, от номера 1 до последнего – FFFF. Вновь создаваемым или пополняемым файлом выделяются ближайшие к началу элементы FAT, но еще свободные и соответствующие им кластеры на МД. При создании файла формируется элемент в каталоге, содержащий информацию:

- имя файла;
- расширение (тип);
- атрибуты (только для чтения, архивный, скрытый, системный);
- дата создания; время создания;
- длина файла в байтах (для подкаталогов нулевая длина);
- номер начального кластера, который выделен для файла.

Кластеры, выделяемые при создании файла, могут находиться не в смежных областях МД, но все они соединены в цепочку, информация о которых содержится в FAT.

Z:\ - корневой каталог



Каждый диск хранит свою файловую структуру. Имеется единственный корневой каталог (\), в который могут входить другие каталоги и файлы.

Для ускорения доступа к файлу древовидной файловой системы необходимо вместе с основным именем указать маршрут его поиска по каталогам.

Каталог – специальный файл, в который регистрируются другие файлы.

При поиске файла на МД ФС обращается к элементу каталога и считывает номер начального кластера. Затем обращается к FAT.

FAT В ячейке таблицы по данному номеру начального кластера содержит номер следующего кластера из цепочки, принадлежащей данному файлу.

Так последовательно один за другим раскручиваются все кластеры для данного файла, конечный содержит код FFFF. Каждому кластеру соответствует физический адрес на МД, поэтому файловая система раскручивает всю цепочку кластеров из FAT, передает управление драйверу МД, который осуществляет считывание информации из файла.

FAT хранится в двух идентичных экземплярах. После завершения работы с файлом или МД обе копии обновляются.

Для ускорения работы при обращении к файловой системе FAT загружается в КЭШ – память, т.е. при активной работе с файлами, информация из FAT Считывается по обращению к Кэш- памяти. По завершении работы обе копии обновляются и FAT выгружается.

Все операции над файлами, также открытие и закрытие файлов доступны перед IRQ –я верхнего уровня 20h-3Fh (кроме 21h-23h).

Чтобы воспользоваться услугами файловой системы программно необходимо выдать соответствующие IRQ, ех, 21h и установить необходимые аргументы в регистрах МП.

Получив управление, файловая система идентифицирует запрос последовательность обращений к драйверу. Драйвер обеспечивает непосредственное управление устройства по выполнению требуемой операции. Через порт ввода/вывода, выдавая последовательность команд, а также принимая и анализируя результаты их выполнения. После завершения операции драйвер возвращает управление и результаты своей работы ядру файловой системы, которая передает результаты программе, выдавшей запрос.

Файловая система MSDOS реализует операции ввода/вывода синхронно, то есть выполнение программы приостанавливается до завершения операции. MSDOS поддерживает работу с FAT16 WINDOWS95, 98, FAT32.

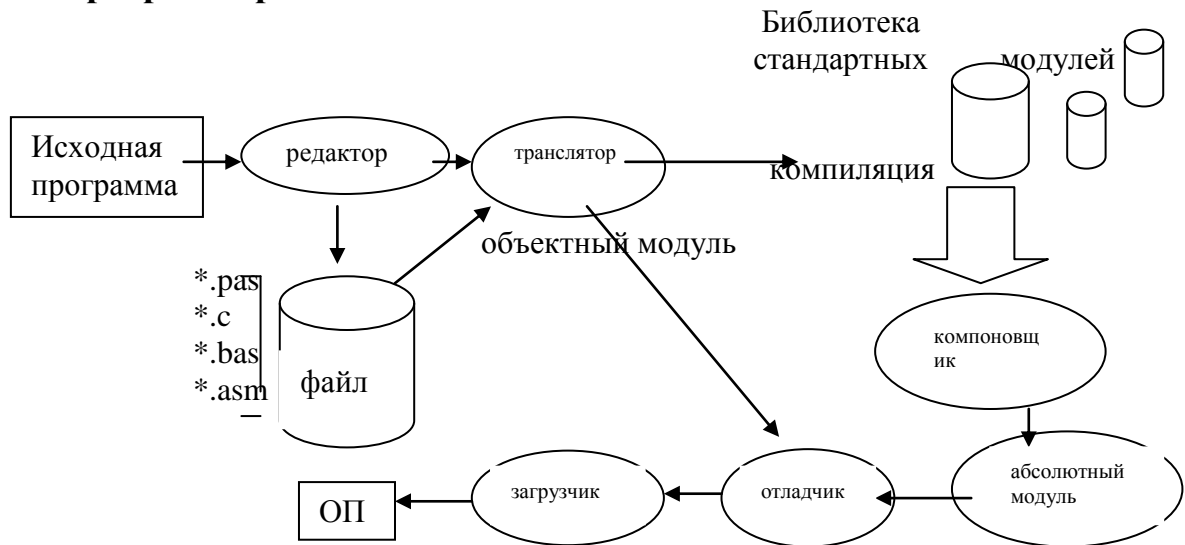
Монитор обслуживается файловой системой не через IRQ-я, а через память в виде адаптера непрерывно. В каждом сеансе работы формируется логический видеобuffer. Все запущенные на выполнение процессы имеют свою структуру в памяти видеобufferа. При необходимости можно переключиться между задачами, при этом активный процесс в данный момент имеет нулевую страницу видеопамати и отображается на экране.

Файловые устройства:

Винчестер HDD, FDD, CD-ROM, CD-RW, Стример – блочного типа. Не файловые устройства все остальные – символьные требуется указать для работы с символьными устройствами в MSDOS подключение драйвера данного устройства device.

В каждом сеансе работы формируется логический видеобuffer. Все запущенные на выполнение процессы имеют свою страницу в памяти видеобufferа, при необходимости можно переключаться между задачами. При этом активный процесс в данный момент имеет нулевую страницу видеопамати и отображается на экране.

2.4.3 Система программирования. Этапы прохождения задачи в системе программирования.



Этапы прохождения задачи в системе программирования:

1. Исходный текст программы на каком-либо языке программирования вводится с помощью текстового редактора в ОП ПК после команды сохранить save формируется на МД файл, расширение соответствует типу файла, т.е. указывает на язык программирования;
2. Набранный текст программы должен быть переведен в машинные коды перед выполнением программы. *Транслятор* – программа переводчик исходного текста программы в машинные коды, а также проверяющий наличие синтаксических и соматических ошибок;
Различают трансляторы:
 - компилятор – компилирующего типа или компилятор – проверяет исходный модуль программы на наличие синтаксических ошибок и переводит в машинные коды, если ошибок нет;
 - интерпретирующего (интерпретатор) Interpreter по командно проверяет на наличие ошибок, переводит в машинные коды и выполняет.
 После компиляции получается объектный модуль программы (файл с расширением *.obj);
3. Компоновщик (редактор связи) объединяет объектный модуль программы и объектным модулем процедур и функций из библиотеки стандартных код программ, вызовы которых имелись по тексту исходной программы. Модули из библиотек вставляются целиком в текст исходной программы и каждой команде присваивается относительный адрес (относительно 0). После компоновки формируется абсолютный (загрузочный модуль программы) (расширение *.exe). Если программа выполняется в среде мультипрограммирования ОС, то при компоновке ей присваивают атрибуты:
 - приоритет в системе;
 - раздел в памяти;
 - код защиты;
 - признак выгружаем ости или резидентности;
 - время цикла выполнения программы, если она должна выполняться циклически.
4. Отладчик debugger – программа, которая позволяет выявлять логические ошибки в решении задачи. Программа запускается под управлением отладчика и с помощью точек останова можно просматривать выполнения команд программы, а также

содержимое регистров МП и содержимое переменных, используемых в данной программе;

5. Загрузчик организует загрузку абсолютного модуля программы в ОП, если программа была запущена в режиме отладки, то процессом выполнения ее будет управлять отладчик. Адрес загрузки программы будет передан МП в счетчик адреса команд. МП начинает по командное выполнение программы.