

## Програмно осигуряване на компютърните системи

Компютърът е универсална информационна машина, която може да извършва много и разнообразни дейности – да осигурява счетоводно и административно управление на фирми, да редактира текстова, графична и друг вид информация, да управлява технологични процеси, да се играят игри, гледат филми и др. Някоя друга машина не може да изпълнява такъв широк диапазон от различни дейности. Този широк обхват на приложение на компютърните системи се обуславя от универсалните електронни устройства, от които са изградени и простата система от команди (инструкции), заложили като функционални възможности на централното процесорно устройство. Разглеждан на това ниво, компютърът е апаратна част (Hardware).

Решаването на разнообразните и сложни задачи в КС се осъществява чрез многократно изпълнение на множество елементарни аритметични и логически операции, включени в системата команди на микропроцесора. Всяка елементарна операция или поредица от команди се изпълнява от машината със задаване на определена команда (инструкция). Следователно, за машинното решаване на коя да е задача е необходимо предварително да бъде създадена поредица от подредени и свързани помежду си команди. Такава поредица от систематизирани и логически обвързани команди се нарича **програма**.

За да се реализират всички възможности на КС, са необходими сложни програми, които свеждат сложните информационни задачи до краен брой прости команди и инструкции от системата на CPU. Тази част от компютърната комплектровка се нарича програмна част (Software). Нейното значение за развитието на КС е огромно. В момента разработването на програмни средства за КС представлява мощна индустрия – софтуерна индустрия, която дори надвишава по приходи и заетост хардуерната индустрия.

### 1 Видове програмно осигуряване

Софтуерната част на компютърните системи се нарича **програмно осигуряване**. Програмното осигуряване може да се раздели на две големи групи: системно (general Software) и приложно (application Software) програмно осигуряване.

Системното програмно осигуряване е основен инструмент за управление на КС и осигурява унификация при работа с различни хардуерни системи. То обезпечава многоцелевото използване на КС. Следователно, програмите, включени в тази група, трябва слабо да зависят от конкретните цели, за които се използва КС. Към системното програмно осигуряване се отнасят програмите за организация на изчислителните процеси в различни режими на работа, програмите, контролиращи изпълнението на задачите, програмите за диагностика и коригиране на възникващите неизправности при пренасянето на данните между различните устройства, програмите за контрол на заданията на потребителите и др. Най-общо, системното осигуряване е това, което се доставя на потребителя заедно с КС. Част от това програмно осигуряване е реализирано в самата апаратура на КС, в така наречената ROM памет. Друга част се зарежда при продажбата на КС, но съществува и възможност за промяна и разширение на системното програмно осигуряване от потребителя в процеса на използване на КС.

Системното програмно осигуряване изпълнява задачи, общи за цялата изчислителна система. То осигурява средата, в която функционират приложните програми. Системното програмно осигуряване изгражда инфраструктурата за изпълнение на програмите на потребителя. Основна част от системното програмно осигуряване представлява операционната система. Останалите елементи от системното програмно осигуряване се наричат обслужващи програми (утилити). В голямата си част обслужващите програми са предназначени за изпълнение на действия, необходими за успешното функциониране на КС, но не са включени в операционната система. Те допълват операционната система. Предоставянето на определени функционални възможности на обслужващите програми

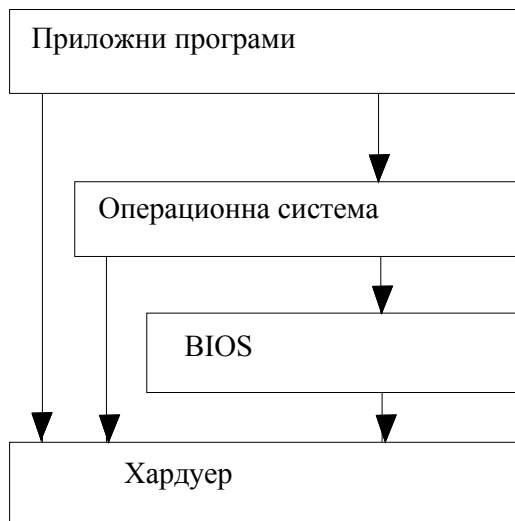
опростява разработката на операционни системи. Това прави системното осигуряване по-гъвкаво и адаптивно към конкретните изисквания за работа на дадена компютърна система.

Приложното програмно осигуряване включва програмите, предназначени за решаване на конкретни задачи на потребителя, произтичащи от специфичните особености при използването на дадената машина. Така например, КС може да бъде използвана за счетоводни разчети или управление на складови наличности, което изисква машината да разполага със специализирани счетоводни програми. Ако КС се използва за подготовка и съхраняване на документи или подготовка на обобщени справки и разчети за фирмената дейност на дадена компания са необходими програми с общо предназначение каквито са например Office пакетите (съдържащи текстообработващи програми, електронни таблици и др.). Към приложното програмно осигуряване можем да отнесем и програмните системи за управление на бази данни, настолните издателски системи, програмите за обработка на графични обекти, мултимедийните приложения, интегрираните работни среди за създаване на приложни програми и др.

## 2 Системно програмно осигуряване

### 2.1 Трипластов модел на КС

Една от най-важните задачи на системното програмно осигуряване е достъпът до хардуерните елементи на КС. Това не може да се извърши директно от програмата на потребителя. Например, не може да се осъществи директно обръщение на микропроцесора към видеоконтролера на КС. Вместо това програмата може да използва ROM – BIOS и операционната система (DOS, Windows), които да подготвят правилна заявка за използване на това устройство. Тези програмни системи се явяват като посредници за достигане на хардуерните елементи. ROM BIOS-ът и операционната система са създадени именно за тази цел – да управляват достъпа до хардуера. Взаимодействието между апаратната част на КС и приложните програми с използване на системното програмно осигуряване може да се представи посредством трипластов модел, както е показано на фиг. 3.1



Фиг. 3.1. Трипластов модел на компютъра

Моделът показва, че най-близо до хардуера се намира BIOS системата, която непосредствено комуникира с апаратните елементи и другите системи се свързват с хардуера посредством BIOS. Следващият слой в структурата на КС се явява операционната система, а най-горният слой съставят приложните програми. Те комуникират с хардуерните елементи чрез операционната система и BIOS. В архитектурата на КС е предвидена възможност и за директно свързване на приложната програма с апаратните елементи, но тази връзка се

използва в случаите, когато се изпълняват специални действия, които операционната система и BIOS не могат да осигурят.

## 2.2. BIOS (Basic Input Output System)

BIOS е термин, описващ всички драйвери на компютърната система, които работят съвместно и оформят интерфейса (връзката) между хардуерните елементи и софтуера. Една част от BIOS-а се разполага в ROM чип, който е постоянна памет само за четене. Това е същинската част на BIOS, но не е цялата. BIOS-ът включва също ROM чипове, намиращи се на разширителните карти (платки), както и допълнителни драйвери, които се зареждат при стартиране на операционната система.

Частта, която се намира в ROM чипове на дънната платка и на някои разширителни карти, понякога се нарича 'фърмуер' (*firmware*) –наименование давано на софтуер, съхраняван в чипове, а не на твърдия диск. Това понякога създава илюзия, че BIOS-ът е хардуерен компонент.

BIOS – е съвкупност от програми, които са необходима за работа на компютъра по всяко време от използването му. Част от тези програми поемат управлението на компютъра при неговото стартиране и организират зареждането на операционната система. Тази част е записана в постоянна памет ROM на дънната платка и се вгражда в компютърната система при асемблирането ѝ.

При стартиране на КС, когато все още няма заредени никакви програми и драйвери за връзка с хардуерните устройства, ROM BIOS програмите са единствените, които се намират в паметта, и управлението автоматично се предава в определен адрес от програмното пространство на BIOS. Той подготвя системата за работа и осигурява връзките с периферните устройства.

Преди години, когато РС компютрите работеха само под управлението на DOS, ROM BIOS беше напълно достатъчен за работа на системата. Не бяха необходими допълнителни драйвери, тъй като BIOS-ът на дънната платка разполагаше с всичко необходимо за управление на устройствата на компютърните системи. Той обикновено включваше драйвери за всички основни компоненти, като клавиатура, флопидискови устройства, харддиск, серийни и паралелни портове и други. Когато системите станаха по-сложни, в BIOS-а на дънната платка липсваха драйвери за доста нови хардуерни устройства, като например видеокарти, CD-ROM устройства, SCSI харддискове, USB портове и други.

Вместо да се разработват дънни платки с нов BIOS, поддържащ новите устройства, по-лесно и практично се оказа новите драйвери да се копират на системния харддиск, а операционната система да ги зарежда по време на своето стартиране. Това е начинът, по който се поддържат повечето CD-ROM устройства, звукови карти, скенери, принтери. Тъй като тези устройства не е необходимо да са активни по време на стартиране на компютърната система, драйверите им не е необходимо да се намират в ROM памет на BIOS.

Някои драйвери обаче трябва да са активни по време на стартиране на системата. Например, как могат да се заредят операционната система и другите драйвери, ако драйверите на харддиска не са в паметта на компютъра (в ROM паметта)? Очевидно е, че драйверите на харддиските трябва предварително да са заредени в ROM, който може да се намира или на дънната платка или на адаптерната (разширителна) платка за диска.

Как може да се види нещо на екрана на монитора, ако видеокартата не разполага с набор от драйвери в ROM? Решението на проблема отново е в осигуряване на подходящи драйвери за видео в ROM паметта. Не е особено практично драйверите за отделните устройства да се записват в ROM паметта на дънната платка защото съществуват множество различни устройства и трябва в ROM-а на дънната платка да се записват множество драйвери. Вместо това инженерите от IBM намират по-добро решение още в първите РС компютри. ROM-ът на дънната платка е проектиран така, че да сканира слотовете в системата и да търси адаптерни карти с разположени на тях ROM-чипове. Ако се открие

карта с ROM, записаният в него софтуер се изпълнява по време на първоначалната фаза на зареждане, преди зареждането на операционната система.

Тъй като ROM-базираните драйвери се поставят директно на разширителните карти, не е нужно да се променя ROM-а на дънната платка за да има поддръжка на нови устройства, особено на такива, които не трябва да са активни при пускане на системата. Някои разширителни (адаптерни) карти почти винаги имат ROM драйвери:

- Видеокарти. Всички видеокарти разполагат с BIOS драйвери;
- SCSI адаптри за твърди дискове. Тези, които поддържат зареждане от SCSI харддискове или CD-ROM устройства имат ROM BIOS драйвери.

- Мрежови карти. Картите, поддържащи зареждане директно от файлов сървър, имат така наречения зареждащ (boot) ROM или IPL (Initial program load – първоначално зареждане на програми) ROM. Това позволява PC компютри да се конфигурират в мрежата като бездискови работни станции – наричат се още Net PC компютри или умни терминали.

- Карти за надграждане на IDE или на флопидисковия контролер. Това са карти, които позволяват да се свързват повече нормалния брой устройства. Тези карти трябва да имат собствен BIOS, за да направят тези устройства зареждаеми.

В BIOS има подпрограми и функции, които управляват прехвърляне на данни между централния процесор и външните устройства. Прехвърлянето се организира на физическо ниво. Например, записването на данни върху Hard диска се извършва, като се формират управляващи импулси за запис с определена магнитна глава (дисква повърхност), писта и сектор на магнитната повърхност от диска. Тези управляващи импулси се предават на контролера на диска, който позиционира четящите глави на определеното място.

BIOS – ът може да се разглежда като пласт, който покрива хардуера и предлага функционални възможности за достъп до следните устройства: Видеоконтролер, RAM памет, Флопи и харддискове, Паралелни и серийни портове, Клавиатура, Мишка, Часовник за реалното време и др.

Въпреки че може да се избегне използването на BIOS и директно да се осъществи достъп до всички устройства, желателно е да се използват функциите на BIOS, защото те са стандартизирани и съществуват във всеки PC. Стандартизацията на функциите на BIOS позволява една програма, написана за една КС да се изпълни безпроблемно на друга КС. Независимо дали системата съдържа 20 MB или 20 GB диск и дали е произведен от IBM или Maxtor, функциите му са идентични и програмите се изпълняват по един и същ начин. Поради тази причина PC са толкова популярни и разпространени.

### **Архитектура на BIOS**

BIOS се намира в ROM паметта, което я прави резидентна програма (намираща се в оперативната памет). Тя се съхранява на най-високото място в адресното пространство на основната памет (1 MB основна памет при конфигурацията на операционната система DOS) – адрес F000H. Известно е, че 1 MB памет може да се адресира с 20 битово двоично число (5 цифрено шестнадесетично число). Адресното пространство за 1 MB памет е от 0 до FFFFH. Обикновено BIOS се разполага от адрес F000H до FFFFH. В зависимост от наличната памет и версията на BIOS началото може да е различно (E000H или C000H).

В компютърните системи се разполага определено количество RAM памет, адресирана със същите адреси както адресите, на които се намира BIOS. Тъй като не е разрешено дублиране на адресите на клетки от паметта, ROM BIOS държи скрита тази памет от операционната система и приложните програми. Тя се нарича скрита RAM памет. Много BIOS системи копират ROM BIOS кода в скритата памет, от която BIOS данните се достъпни за операционната система и приложните програми. Това подобрява скоростта на работа на PC, защото шината за данни на скритата RAM памет е 16 битова (32 битова е новите системи), а тази на ROM BIOS е 8 битова.

### **Харддискови параметри**

BIOS често има проблеми при комуникацията с многото твърди дискове, намиращи се на пазара. Това се дължи на различните параметри за комуникация. BIOS трябва да познава

броя на четящите глави, пистите (пътеките) и секторите, които съставят пространството за съхраняване на информация в твърдия диск, служебните писти и някои други данни. Тези параметри на твърдия диск се уточняват чрез специална програма SETUP от системата на BIOS.

Данните за твърдия диск и другите данни за системата се записват в специална памет, поддържана от батерията на компютъра (CMOS памет)

### **Програма SETUP**

Програмата SETUP позволява на потребителя да конфигурира някои от елементите на компютърната система посредством параметрите на BIOS системата. Тези елементи са: системна дата и време, типове на устройствата, програмно изключване на устройства, разширение на паметта и др. Голяма част от SETUP програмите поддържат и управление на основната честота на системната шина и честотата на микропроцесора и др.

BIOS е системата, чрез която може да се въведе защита от неправомерен достъп до компютърната система чрез използването на парола. В този случай паролата се записва в паметта, поддържана от вътрешната батерия на КС.

### **POST процедура**

За компютърните системи с микропроцесори Intel началото на работа става чрез задаване на управлението в клетка от паметта с адрес FFFF0. Тази клетка е от BIOS и съдържа команда за предаване на управлението на процедура (програма), която извършва проверка на системата и инициализация на хардуера. Тази процедура се нарича POST (Power On-Self Test).

Процедурата се състои от многото тестове за проверка на наличните устройства (процесор, памет, контролери на прекъсванията, DMA, контролери за устройствата). Ако възникне грешка, POST формира съобщение за грешка върху дисплея и подава звуков сигнал.

### **Възможности за настройки чрез BIOS**

Посредством програмата SETUP на BIOS могат да се правят различни настройки в зависимост от типа на BIOS системата и компютърната система. Някои от най-често използваните настройки в съвременните BIOS системи са:

- устройства за първоначално зареждане на операционната система (A, C, CD-ROM,...). По този начин могат да се използват различни операционни системи;
- Задаване на основна видеокарта, ако са монтирани 2 видеокарти (AGP и PCI);
- Деактивиране на портове (ако даден порт не се използва да се освободи IRQ линия. Може да се деактивира и порта PS/2 на мишката, ако тя е монтирана в някой USB порт;
- Конфигуриране на честотата на централния процесор;
- Намаляване на броя на тестовете на КС при първоначално пускане;

## **2.3 Операционни системи**

Съвременните КС представляват съвкупност от сложни електронни устройства, свързани в единна система. Голяма част от управлението и взаимодействието на тези устройства е възложено на специална програмна система (програма), наречена операционна система.

В КС често се налага изпълнението на няколко взаимно конкуриращи се дейности. Например, в определен момент може да се печата някакъв документ, да се модифицира (редактира) друг документ и да се създава графическо изображение. За изпълнението на тези задачи е необходима висока степен на координация, гарантираща, че извършваните действия няма да оказват влияние едно на друго и връзката между заданията ще бъде ефективна и надеждна. Аналогични проблеми възникват, когато няколко машини се обединяват в едно цяло, образувайки компютърна мрежа. Решаването на тези проблеми се извършва от операционната система.

### **7.2.3.1 Развитие на операционните системи**

В периода 1940 – 1950 година електронно-изчислителните машини (ЕИМ) са били недостатъчно гъвкави и ефективни. Изпълнението на дадена програма е изисквало специална подготовка: позициониране на лентови устройства, комплектоване на перфокарти в четящо устройство, установяване на превключватели на управляващо табло и др. Пускането на една програма в тези условия се нарича задание (Job) и се извършва самостоятелно. За да се ползва машината от различни потребители, се прави специално разписание за резервиране на машинно време. По време на изпълнение на едно задание машината е напълно в разпореждане на това задание.

Първите опити за усъвършенстване процеса на подготовка и изпълнение на програмите на потребителите се отнасяли до ускоряване на прехода от едно задание към друго. За целта е трябвало потребителите да бъдат отделени от оборудването (управлението на електронно-изчислителните машини), за да се избегне постоянният поток от хора намиращи се помещенията с управляваща апаратура. Била въведена длъжност оператор на ЕИМ, чиято задача била да изпълнява всички операции, свързани с оборудването при изпълнение на дадена програма. Всеки потребител е трябвало да представи на оператора програмата си и заедно с нея необходимите данни и инструкции за изпълнението ѝ, а след това да чака за получаване на резултатите от изпълнението на програмата. Операторът зареждал програмата на потребителя в основната памет на ЕИМ и от там операционната система е можела да започне изпълнението на програмата. Когато е било необходимо в едно задание да се изпълнят няколко последователни обработки, потребителят е подготвял последователните задачи в един общ пакет. Това е било началото на така наречената пакетна обработка на заданията – метод за изпълнение на заданията, посредством предварителното им обединяване в единен пакет, който по нататък се изпълнявал без взаимодействие с потребителя. Заданията, очакващи своето изпълнение в масовата памет, са образували така наречената ‘опашка’ на заданията.

В ранните системи ЕИМ, при пакетната обработка всяко задание се е съпровождало с редица инструкции, описващи необходимите стъпки за подготовка на заданието за изпълнение. Тези инструкции са се записвали посредством специален език за управление на заданията (JCL – Job Control Language) и се комплектовали заедно с програмата в един пакет (обикновено перфокарти), който се предавал на оператора. Някои от инструкциите са изисквали вмешателство на оператора, най-вече когато е имало обръщения към някои периферни устройства.

Тъй като в съвременните компютърни системи тези действия са сведени до минимум, езикът JCL се е превърнал по скоро в език за комуникация с операционната система, отколкото с оператора на машината. Сега длъжността оператор на КС е ненужна, но някои от задълженията по организацията на процесите в компютърните мрежи са пренесени върху системните администратори.

Главният недостатък на традиционните системи за пакетна обработка на заданията е, че потребителят е лишен от възможността за взаимодействие с програмата. За избягване на този недостатък са разработени нови операционни системи, които позволяват при изпълнението на дадена програма да се осъществява диалог с потребителя, който е отделен от системата и работи на терминал или работна станция. Такъв режим се нарича диалогов режим на работа с компютърната система. За такъв режим на работа е необходимо действията, които изпълнява машината, да се координират с протичащите по време на обработката процеси. Тази координация на действията на машината се нарича обработка в **реално време**.

Когато КС изпълнява заданието само на един потребител, обработката на заданието в реално време не представлява проблем. Тъй като КС са доста скъпи съоръжения, се налага една машина да обслужва няколко потребителя. Обработката в реално време в този случай става доста по-трудна, тъй като няколко потребителя едновременно могат да поискат достъп до машинните ресурси и тогава би се получил конфликт. Решението на този проблем се състои в разработката на операционна система, която да организира постоянно редуване на

изпълнението на отделни части от различните задания с помощта на процес на разделяне на времето. Този метод се заключава в разделяне на машинното време на интервали (*кванти*) с последващо ограничаване на времето за непрекъснато изпълнение на програмите до един квант. Когато изтече един квант за дадена програма, заданието на тази програма се сменя от изпълнение и се изпълнява ново задание. Когато се изредят всички задания, отново се връща изпълнението на първото задание. При бързо редуване на заданията, остава впечатление за едновременно изпълнение на всички задания.

В съвременните КС разделянето на времето за изпълнение се използва както в многопотребителските системи, така и в системите с един потребител, но в така наречения многопрограмен режим. Този начин на организация на работата на КС е доста ефективен, въпреки неизбежната загуба на машинно време за превключване на системата за работа с различните задачи. Това време се компенсира в много голяма степен от времето на престой на процесорното устройство при операциите с периферните устройства. По този начин дори се повишава общата ефективност на КС, тъй като престоите на микропроцесора са много по-малки.

### 2.3.2. Архитектура на операционните системи

Частта от ОС, която обезпечавя интерфейса с потребителя често се нарича **обвивка** на операционната система. Обвивката на съвременните КС се изпълнява посредством **графически интерфейс на потребителя (Graphical User Interface – GUI)**, при който обработваният елемент (файл, програма) се представя на екрана на монитора с помощта на малки картинки, наричани икони. Системите, използващи GUI, позволяват на потребителя да въвежда команди, показвайки съответната икона с помощта на посочващото устройство (мишка). В по-старите операционни системи (DOS) обвивката осъществяваше диалога с потребителя посредством текстови съобщения, въведени от клавиатурата и изписвани върху екрана.

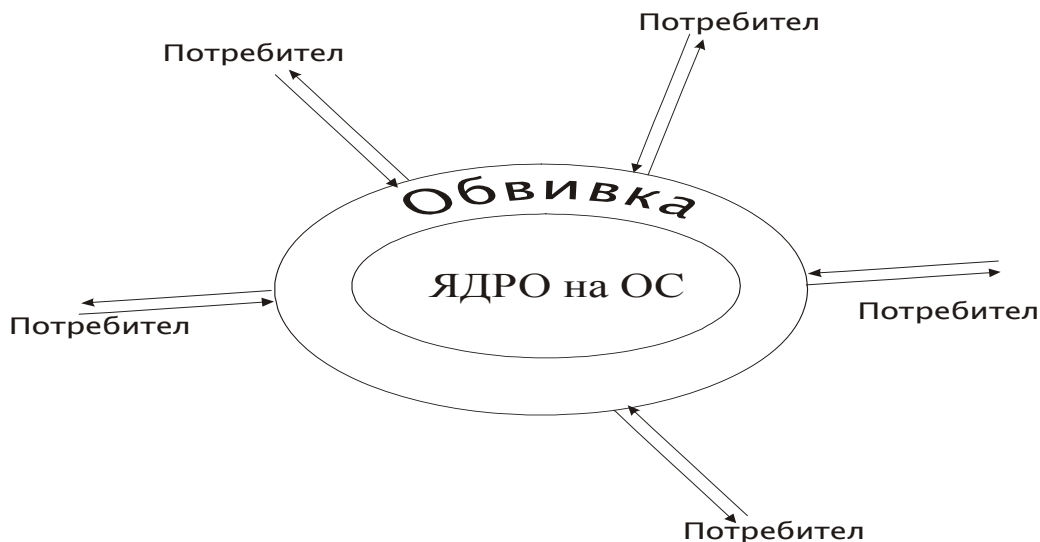
Въпреки че обвивката на ОС играе важна роля за функционирането на КС, тя все пак е само интерфейс (връзка) между потребителя и програмните елементи, които изпълняват функционалните действия на ОС. Различието между обвивката и вътрешните компоненти на ОС се обуславя и от възможността на съвременните ОС да предоставят на потребителя за избор на най-подходяща и удобна за работа обвивка. Например, системата UNIX дава възможност на потребителя да избере една от обвивките, обозначени като Bourne, C или Korn. В ОС Windows е предвидена възможност за настройка на отделни елементи от обвивката, така че потребителят може да конфигурира свой, собствен стил на обвивка на ОС.

Главният компонент на съвременните графически обвивки се явява програмата за управление на прозорци, която разпределя отделните блокове от пространството на екрана на монитора, наричани прозорци и назначава за всяко приложение (програма) отделен прозорец. Когато дадено приложение трябва да изобрази нещо на екрана, то съобщава за това на програмата за управление на прозорците. Тази програма помества изображението, което генерира приложението в неговия прозорец. По същия начин, когато потребителят натисне клавиша на мишката, програмата за управление на прозорците определя положението на показалеца и уведомява съответното приложение за това действие на потребителя.

Съвкупността от вътрешните компоненти на операционната система, които изпълняват основните функции, поддържащи компютъра в работно състояние, обикновено се нарича **ядро** на ОС (Фиг.3.2). Основните елементи на ядрото на операционните системи са:

- Подсистема за **управление на файловете**. Една от най-важните компоненти на ядрото на ОС се явява **програмата за управление на файловете**. Голяма част от времето на КС се изразходва за въвеждане и извеждане на данни към и от външната памет. Това се осъществява от системата за управление на файловете (данните). Тя осигурява достъпа до програмите и данните от външната памет; надеждно и компактно съхраняване на

информация на външни устройства; организира разпределението на пространството върху носителите на външните устройства. За да се осигури надеждно опериране с данните и програмите се въвежда понятието **файл**. Файлът е савкупност от данни, разпределени в отделни групи, наречени записи. Цялата информация, което се съхранява на външни носители се оформя като файлове. Файлове са текстовете на програмите, документите, които се редактират с текстови редактори, графичните изображения и др.



Фиг 3.2 Компоненти на операционната система

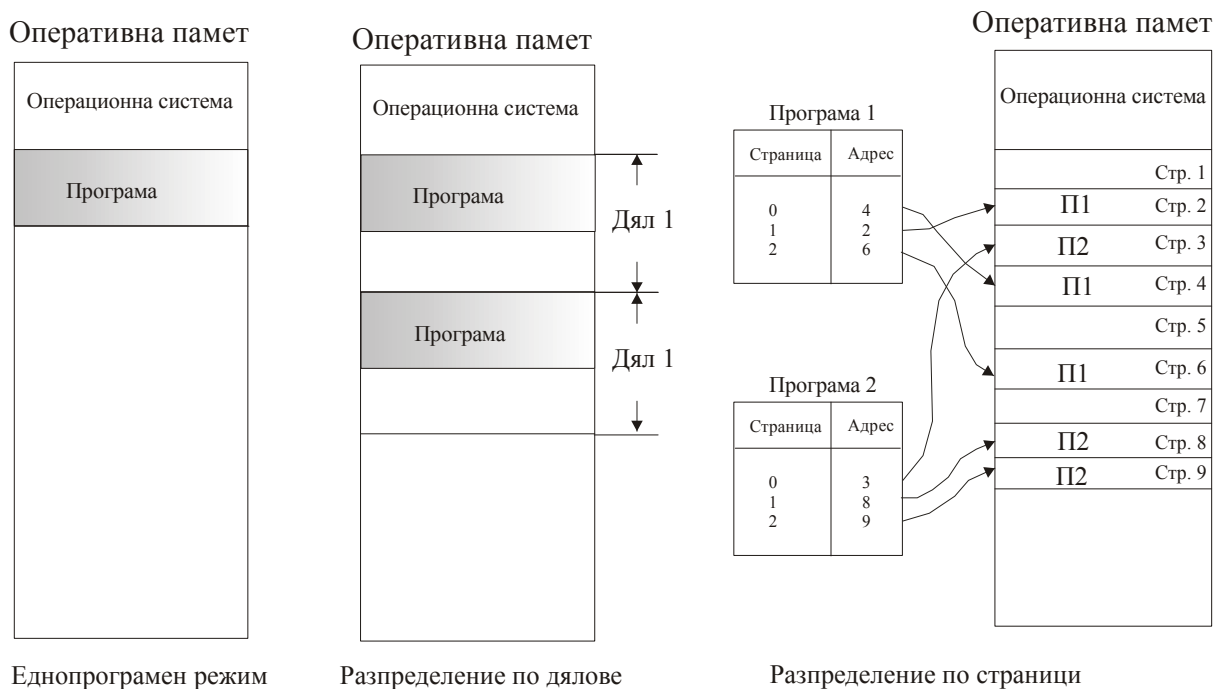
Организацията на информацията във файла се определя от типа му и програмата, която го създава. Всеки файл се идентифицира с определено наименование, а за всички файлове върху запомнящото устройство се създава специален каталог (справочник).

- Подсистема за **управление на паметта (ПУП)**. Тя извършва координация при работа на КС с оперативната памет. В режим на работа с много задачи или потребители тази програма играе съществена роля за управлението на процесите и използването на паметта. В такава ситуация в основната памет са разположени множество програми и блокове от данни, при което всяка програма трябва да има собствен дял от паметта. В процеса на работа потребността от памет за различните програми се променя и управляващата тези процеси компонента от ОС трябва да удовлетворява възникващите потребности на програмите.

Задачата на програмата за управление на паметта се усложнява значително, когато обема на паметта в КС е недостатъчна. В тези случаи Програмата за управление на паметта може да създаде илюзия за увеличен обем памет чрез преместване на програми и данни от оперативната памет в паметта на друго устройство (твърд диск). Този вид 'увеличена' памет се нарича виртуална памет. Например, ако изпълнението на дадена програма изисква 64 МВ памет, а КС има на разположение само 32 МВ. За да създаде илюзия, че паметта е по-голяма, ПУП разделя общия обем на паметта на елементи, наречени страници и ги съхранява във външна памет, докато те не се използват в даден момент и ги прехвърля в ОП, когато те стават активни (трябва да се използват). ПУП държи в ОП тези страници, които са активни в даден момент от време и ги заменя с други, когато не са необходими.

Съществуват различни модели за разпределение на паметта: разпределение на паметта при еднопрограмен режим, фиксирано разпределение на паметта по дялове; разпределение на паметта по страници и др. На фиг. 3.3 е представен графичен модел на паметта при трите типа разпределение.





Фиг. 3.3. Разпределение на паметта

-Подсистема за **управление на процесите**. Във всеки момент от време, независимо колко програми са заредени едновременно в паметта, компютърът изпълнява само един програмен процес (програма). Така е защото системата има само един микропроцесор. Съществуват и многопроцесорни системи, но за тях операционните системи имат друга архитектура.

Една от най-важните концепции при съвременните КС е разграничението между понятието програма и действията, свързани с нейното изпълнение. Програмата представлява статичен набор от команди, докато процесът представлява съвкупност от действия, които се изменят във времето. Текущото състояние на работата по изпълнението на дадена програма се нарича състояние на процеса. Състоянието на процеса включва текущата позиция от изпълняемата програма (стойността на брояча на команди), а така също и съдържанието на основните регистри на централния процесор и на клетките от паметта, в които се намират данните, които се обработват в момента. Казано образно, състоянието на процеса е моментна снимка на състоянието на машината в дадения момент от време. За да се подчертае различието между програма и процес, може да се отбележи, че една програма може да бъде свързана едновременно с няколко процеса. Например, в многопрограмен режим двама или повече потребители могат да обработват с една и съща програма повече от един документ, което означава, че са инициирани повече от един процеса с тази програма. В този случай в оперативната памет се намира само едно копие на обработващата програма, но с нея се изпълняват няколко процеса.

Координацията при изпълнението на процесите изисква от подсистемата за управление на процесите да гарантира на всеки процес необходимите му ресурси (достъп до периферни устройства, оперативна памет, машинно време на централния процесор и др.). Освен това, независимите процеси не трябва да си влияят един на друг, а за тези, които трябва да обменят информация помежду си да се осигури такава възможност.

Задачите, свързани с координацията на процесите, се осъществяват от програми, наречени **планировчик** и **диспетчер**. Планировчикът води специален дневник на процесите, присъстващи в компютърната система, добавя нови сведения и регистрира нови процеси, затваря записите за процеси, които са завършили. Информацията, която създава и управлява планировчикът, се организира във вид на таблица на процесите.

Диспетчерът определя стратегията за редуване на изпълнението на различни програмни процеси и управлението на тези процеси. В системите с разделяне на времето задачата се решава чрез разделяне на времето на процесора на кратки интервали, наречени **кванти** (обикновено с продължителност 50 милисекунди). При изтичането на това време се извършва принудително превключване на централния процесор от един процес към друг. Всеки път, когато на даден процес се предостави време (квант), диспетчерът включва веригата на таймера за отмерване на времето за работа.

Понякога използваното от даден процес време може да завърши преди да е изтекъл даденият му за ползване квант време. Обикновено в тези ситуации се прилага система на приоритети. Ако една програма работи и в определен момент се налага прекъсване (например заради обръщение към външни устройства), което е свързано с изчакване, операционната система натоварва микропроцесора с изпълнението на друга програма, която има най-висок приоритет в този момент. Обикновено, входно-изходните операции се извършват от устройства, които имат собствени микропроцесорни устройства или се използва контролерът за директен достъп до паметта (DMA).

- Подсистема за **управление на периферните устройства**. Ефективността на цялата компютърна система се определя от оптималното разпределение на натоварването на отделните устройства. Операционната система притежава функционални възможности за управление на работата на периферните устройства и оптимизиране на достъпа до тях. Към системата за управление на периферните устройства се отнася и наборът от **драйвери** на устройствата свързани в КС. Това са програми (елементи от програмното осигуряване), чрез които се осъществява връзката с контролерите на устройствата или непосредствено със самите устройства. Всеки драйвер се разработва специално за конкретния тип устройство (принтер, дисков контролер, монитор и др.). Той преобразува постъпващите заявки за използване на съответното устройство в команди за изпълнение на отделните физически операции от устройството. Посредством драйверните програми операционната система може да се настрои за работа с различни по тип устройства.

#### - Модел “клиент/сървър”

Различните компоненти на операционната система обикновено работят като отделни процеси, конкуриращи се в системите с разпределение на времето. За координация на действията между отделните процеси операционните системи се разработват в съответствие с модела “клиент/сървър”. Съгласно този модел всеки компонент влиза в ролята на клиент, изпращащ запитване до друг компонент или в ролята на сървър, отговарящ на запитване постъпило от друг компонент (клиент). Например, програмата за управление на файлове функционира като сървър, предоставящ достъп до файловете в съответствие със запитвания, постъпващи от различни клиенти. Разработеното по този модел взаимодействие между процесите вътре в операционната система предполага постъпване на запитвания от процеси, изпълняващи роля на клиенти и предоставяне на отговори от други процеси, играещи ролята на сървъри.

При разработката на програмно обезпечаване съблюдаването на принципа на модела “клиент/сървър” позволява точно да се определи ролята на отделните елементи. Клиентът изпраща запитвания към сървъра и очаква отговор, а сървърът обслужва постъпващите запитвания и отговаря на клиентите. Ролята на сървъра не зависи от това дали клиентаът функционира на този компютър или на някаква отдалечена машина свързана в обща мрежа. Различия съществуват в програмното обезпечаване, използвано за взаимодействие между отделните компютри, но не и във взаимодействието между клиента и сървъра. В резултат, ако компонентите на някоя програмна система бъдат организирани като клиенти и сървъри те могат да изпълняват своите функции независимо дали функционират на една машина или на различни машини, разделени с голямо разстояние. Затова, ако програмното обезпечаване, образуващо най-ниското ниво на системата, представлява средства за обмен на запитвания и отговори на тях, сървърите и клиентите могат да бъдат разпределени по машини с каква да е конфигурация.

Стремежът да се установи универсална система за трансфер на съобщения, която може да поддържа разпределените системи в компютърните мрежи се явява цел на редица стандарти и спецификации известни под наименованието CORBA (Common Object Request Broker Architecture – архитектура на брокери на запитвания за общи обекти). Спецификацията CORBA включва система стандарти, регламентиращи мрежовите взаимодействия на елементите на програмното обезпечаване (клиенти и сървъри).

### 2.3.3. Организация на компютърни мрежи

Едно от най-важните изисквания към съвременните компютърни системи е необходимостта от съвместно използване на информацията и ресурсите на различни машини. Затова компютрите могат да се обединяват в групи свързани помежду си машини, наричани компютърни мрежи. В миналото основната концепция беше една централна голяма машина да обслужва множество потребители (терминали, малки компютри). Сега по-често се използва концепцията множество малки компютърни машини да се обединяват в мрежа, в която потребителите ползват съвместно ресурсите, разпределени в цялата мрежа – принтери, скенери, програмни пакети, големи дискове или информация.

Голяма част от проблемите по координацията на работата при създаването на компютърните мрежи се отнася до частта от операционната система, която управлява обмена на данни между компютърните системи в мрежата. По същество програмното осигуряване за управление на мрежите може да се разглежда като мрежова операционна система и е естествено разширение на стандартните операционни системи за управление на една компютърна машина.

Всяка компютърна мрежа принадлежи към някоя от следните основни категории: локална изчислителна мрежа (LAN, Local Area Networks) и глобални изчислителни мрежи (WAN, Wide Area Networks). Локалната мрежа, като правило се състои от няколко компютъра, намиращи се в една сграда или в комплекс от сгради. Например, компютрите използвани в даден университет или в завод могат да се свържат в една LAN мрежа. Глобалната компютърна мрежа свързва компютри, намиращи се на голямо разстояние един от друг, например компютри, намиращи се в различни части на даден град, страна или света. Основната разлика между локалните и глобални мрежи се състои в различните технологии използвани за свързване на отделните компютърни системи.

#### **Internet**

Ако се свържат няколко съществуващи компютърни мрежи в една обща мрежа се получава мрежа от мрежи. Най-известната такава мрежа е глобалната мрежа **Internet**. Тя възниква през 1973 година с разработване на програма започната от американската агенция DARPA, извършваща изследвания за министерството на отбраната на САЩ. Целта е била да се разработят средства за свързване на разнообразни компютърни мрежи, които да работят надеждно като единна мрежа. Сега Internet е глобално обединение на голям брой локални и глобални мрежи, включваща милиони компютри по целия свят. Мрежите в Internet са свързани с помощта на специални машини, наричани маршрутизатори (шлюзове). Тяхната роля е да управляват пренасянето на информация между отделните мрежи в глобалната мрежа.

Internet може да се разглежда като обединение на мрежови клъстери (обособени глобални или локални мрежи), които се наричат домейни (**domain**). Всеки домейн обикновено се състои от мрежи, принадлежащи на някоя организация (университет, учреждение, министерство, голяма компания и други). Всеки домейн се явява автономна система.

Адресът на всяка машина в Internet се задава със символен низ от 32 бита, състоящ се от две части: първата част задава домейна, в който се намира машината, а втората определя конкретната машина в домейна. Частта от адреса определяща домейна се нарича мрежов идентификатор и се присвоява на домейна от организацията InterNIC (Internet Network Information Center - Център за мрежова информация в Internet). Това се извършва при

регистрацията на домейна в InterNIC. Тази процедура гарантира, че всеки домейн ще има уникален мрежов идентификатор и ще бъде достъпен от всички домейни свързани в Internet.

Частта от адреса, определящ конкретната машина в домейна се нарича адрес на възела (host). Адресът на всеки възел се задава от локалния администратор на домейна. Адресът на една машина се съхранява в битова форма и изглежда по следния начин: 192.207.177.133. Отделните байтове в адреса обикновено се записват с точка между тях. Тук първите три тройки числа определят мрежовия идентификатор на домейна, а последните - номера на машината в домейна. Представянето на адреса в битова форма е неудобно за използване, поради което InterNIC присвоява на всеки домейн и уникален мнемоничен адрес, който се явява име на домейна. Така например, мнемоничния адрес на домейна на Югозападен университет е **swu.bg**.

Отбелязването на адресите в Internet с използване на точки няма отношение към десетичното представяне на числата. Те специфицират местоположението на дадена машина в сложната йерархическа структура на глобалната мрежа. Когато даден домейн е много голям, локалната администрация може да го раздели на поддомейни и тогава мнемоническите адреси вътре в домейна стават по-дълги (съдържат имената на поддомейните).

#### **2.3.4. Стартиране на операционната система**

Стартирането на ОС се извършва с помощта на специален процес, наречен зареждане на ОС (bootig), който се изпълнява при стартирането на КС. Това се налага от самата организация на работата в КС. Централният процесор е разработен така, че винаги да изпълнява някаква програма. При стартирането на КС, в паметта няма заредена програма и затова е възприета стратегията CPU да се обръща към определен адрес, намиращ се в постоянната памет ROM, където трябва да има постоянна записана програма. От там процесорът трябва да прочете първата команда която да изпълнява. За да има сигурност, че там непременно ще има изпълнима команда, тази област от паметта трябва да е типът ROM памет.

Тъй като тази памет при компютърните системи с общо предназначение е само за начално стартиране на машината, тя обикновено е с малък обем. В други системи, които се използват за управление на машини и прибори (автомобили, електрически уреди), изпълняваните програми са постоянни и постоянната памет (ROM) трябва да е голяма.

В ROM паметта на КС с общо предназначение се намира малка програма за първоначално зареждане (bootstart). Тази програма се изпълнява автоматично, при всяко включване на компютъра. Тя задава на микропроцесора задача да прочете данни и програми, записани на някой от външните запомнящи устройства и да ги прехвърли в основната памет на компютъра (ОП). В повечето случаи тези данни и програми представляват операционната система. След зареждането на операционната система в оперативната памет, програмата за начално зареждане генерира команда към микропроцесора да започне изпълнението на програмите от ядрото на операционната система. По този начин се стартират програмите от ядрото на ОС и управлението на КС се прехвърля към операционната система.

В голяма част от съвременните компютърни системи програмата за начално зареждане е настроена така, че първоначално се опитва да зареди операционната система от флопидисковото устройство и ако не открие дискета в него, зарежда ОС от твърдия диск (Hard disk). Ако във флопидисковото устройство има дискета, но на нея не е записана операционната система (не е системна дискета), програмата за начално стартиране спира работата на системата и издава съобщение за липса на системен диск. Поредността на търсене на операционната система върху външните запомнящи устройства се задава чрез BIOS.