



ШУМЕНСКИ УНИВЕРСИТЕТ

“ЕПИСКОП КОНСТАНТИН ПРЕСЛАВСКИ”

ФАКУЛТЕТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

Катедра “КСТ”

Доц.д-р Станимир Станев

Доц.д-р Иван Цонев

УЧЕБНИ МАТЕРИАЛИ
към курса за обучение по магистърска
програма НУПИТ,
по дисциплините
“Компютърни системи”
“Компютърни мрежи и Интернет”

Шумен
2007

СЪДЪРЖАНИЕ

Глава 1. АПАРАТНА ЧАСТ (ХАРДУЕР) НА КОМПЮТЪРА

Глава 2. ПРОГРАМНО ОСИГУРЯВАНЕ (СОФТУЕР)

Глава 3. КОМПЮТЪРНИ МРЕЖИ И ИНТЕРНЕТ

Глава 4. WEB-ТЕХНОЛОГИИ

Литература

Приложения

Глава 1. АПАРАТНА ЧАСТ (ХАРДУЕР) НА КОМПЮТЪРА

1.1 Основни термини

Думата **информация** произхожда от латинското **informatio**, което означава “сведения, разяснения, изложение”. Информацията е философско понятие, което трудно се дефинира. Негови синоними в различен контекст, са сведения, съобщения, знания, факти, данни и др. В най-общ смисъл информация се наричат различни данни или сведения, които интересуват някого, например съобщение за някакви събития, за някаква дейност. Да се информира в този смисъл означава да се съобщи нещо, неизвестно отпреди.

Информация – сведения за обектите и явленията от околната среда, техните параметри, свойства и състояния, които възприемат информационните системи /живи организми, управляващи машини и др./ в процеса на живот и работа.

Клонът на информатиката, свързан с използване на компютъра за обработка на данни, се нарича **компютърна информатика**. (В този смисъл най-близкото аналогично название на английски език на това направление е **Computer sciences-компютърни науки**).

Информационните дейности са съставна част на всяка човешка дейност. Те са възприемане, осмисляне, запаметяване и предаване. Всяка информационна дейност се моделира като **информационен процес**- абстрактен модел на съответната информация за реална дейност, осъществявана в пространството и времето. Процесите, свързани с определени операции над информацията, се наричат информационни процеси. Съответните на човешките информационни дейности **основни информационни процеси** са събиране, обработка, съхраняване и разпространяване.

Понятието **система** представлява съвкупност от елементи, работещи заедно в определена среда за постигане на дадена цел. Информационната система е система, чрез която данните и информацията се обработват и предават от един потребител към друг, или от една организация към друга. Тази система е предназначена да осигури оптималното решаване на конкретен проблем. Не е задължително да включва в състава

си компютри. Информационната система има входна подсистема, обработваща подсистема, изходна подсистема, средства за обратна връзка и управление.

Даден информационен процес може да се разгледа като композиция на основните .

Управлението е информационен процес, при който един обект, наричан управляващ, оказва с определена цел въздействие върху поведението на друг обект, наричан управляван. Цел на управлението е поддържането на определено състояние или получаване на резултат. **Управляващият обект** въздейства върху управлявания чрез **канал за права връзка**, по който предава данни, наречени управляваща информация. Данните, представляващи информация за състоянието на **управлявания обект**, се предават към управляващия по **канал за обратна връзка**. Върху двата обекта и каналите оказват влияние външни въздействия. Наличието на обратната връзка не е задължително, но тя прави управлението по-гъвкаво и адаптивно. Когато управлявания обект следва предварително изготвена програма за действие, се реализира **програмен принцип на управление**. Машините с този принцип на управление са **автоматични**, ако самостоятелно изпълняват съответната програма или **автоматизирани**, ако изпълняват програмата в диалог с човек.

Представянето на информация за обект или процес върху физически носител, представляващо форма за разпространение на тази информация , се нарича **съобщение**. Съобщението е конкретен израз на определена информация. Възможно е една информация да бъде изразена с различни съобщения. Например съобщенията "15 септември" и "Първият учебен ден" изразяват информация за начало на учебната година. Обратно - едно съобщение може да изразява различна информация. Цифрата "6" може да означава брой хора, оценка, стойност. Тъй като е възможно едно съобщение да изразява различна информация, то информацията може да се разглежда като начин за осмисляне на съобщенията. Установяването на еднаквост между информация и съобщение е резултат на предварителна договореност между предаващия и приемащия дадено съобщение.

Средствата за физическо и знаково моделиране на информацията са разнообразни- радиосигнали, звуци, графични знаци и др.

Сигналът е материален носител на съобщения, средство за пренасяне на информацията в пространството и времето.

В зависимост от начина на изразяване, информацията може да бъде **дискретна и непрекъснатата**. Непрекъснатата се изразява чрез съобщения, които представляват непрекъснати във времето сигнали или обекти в пространството. Например речта и филмите предават информация с непрекъснат характер. Дискретната (не в смисъл на конфиденционална) информация се състои от съобщения, съставени от отделни елементи, разположени в пространството и времето. Например печатният текст предава дискретна информация. Информацията, предадена чрез дадена художествена изложба е дискретна. Но най-често информацията има смесен характер, т.е. тя представлява композиция от непрекъснатата и дискретна информация.

Данните са знакови модели с условно подобие за съответната информация. Прости данни са числата, символите и думите, а сложни- графика, анимация, аудио и видео данни. Данните се интерпретират като информация само когато се разглеждат в съответен контекст. Данните са "сурови" цифрови факти. Те могат да се разглеждат като стойности на величини.

Величината е абстрактен информационен модел,отразяващ дадено свойство на реален обект (процес). Величина трябва да има уникално име, и се характеризира с тип и текуща стойност. Има два вида величини- **константи** и **променливи**. Величината се счита за **определена**, ако се познават нейното име и множеството от стойности,

които тя може да приеме в различни моменти при изпълнението на програмата. Данните са стойностите, които приема величината. Величината е основно средство за изразяване на различни характеристики и свойства на предметите и явленията.

Типът на данните е математическа и информационна структура. Той е основно свойство на данните. Всеки тип определя допустимо множество от стойности на съответните данни и операции, които могат да се извършат над тези данни. С типа на данните са свързани и начинът на представяне на данните в оперативната памет на компютъра.

Чрез създадените от човека технически средства в информационните процеси се обработва не самата информация, а само нейните представяния върху физически носители (знакови модели).

В практиката (даже и на специалисти в компютърния бранш) често се използват изразите “предаване на информация”, “компютърна обработка на информация”, въпреки че строго погледнато в светлината на изложеното до тук става дума за “предаване на данни(съобщения)”или “компютърна обработка на данни”.

Медия се нарича физическа среда, имаща подходящи свойства за съхраняване и разпространяване на съобщения. Медиите са аналогови или дискретни (цифрови). Все по-широко разпространение получават дискретните(цифровите) начини за съхраняване и разпространение на аналогова по своята същност информация. Основната причина за това е по-голямата сигурност и икономичност на цифровите по сравнение с аналоговите носители и медии.

Знанието е осъзната и разбрана съвкупност от информация за даден обект, явление или събитие. То е ключ към доброто решение на даден проблем. Важно е да се използва нужната информация точно и на време.

В компютърните информационни системи за представяне на числовите данни се използват двоични цифри и мерните единици **bit- бит**(кодира се с **b**) и **bite- байт** (кодира се с **B**). Двоичната цифра в един бит е 1 или 0. Един байт се състои от последователно разположени един до друг 8 бита. Използват се и по-големи мерни единици – KB- килобайт, MB-мегабайт, GB- гигабайт, TB- терабайт. Съотношенията между тях са:

1 KB (килобайт) = 1024 B, 1 MB (мегабайт) = 1024(2¹⁰)KB, 1 GB (гигабайт) = 1024(2¹⁰) MB, 1 TB (терабайт) = 1024(2¹⁰) GB.

За да се съхрани една страница печатен текст в паметта, е необходимо приблизително 2 KB, една книга от 500 страници заема 1MB .

Машинна дума са един или няколко байта, които се обработват едновременно от компютъра. **Блок** е множество от битове или байтове, обединени с цел пренасянето или обработката им. **Файл** е взаимосвързана обособена съвкупност от данни под общо наименование.

Съвременните компютри са електронни автоматични устройства, работещи на принципа на програмното управление. Те преобразуват данни (цифрови факти) в информация (организирана използваема форма). При това изпълняват функциите управление,обработка, съхраняване и трансфер на данни.

Реализацията на изчислителния процес в компютрите със съхранявана програма се извършва от техните два главни компонента - апаратната част- **хардуер** (hardware) и програмното осигуряване- **софтуер** (software). Докато в началото на 50-те години от миналия век почти 90% от стойността на тогавашните компютри бе на хардуера, то сега по-голямата част от стойността е на софтуера.

. До средата на 80-те години у нас вместо термина "компютър" се използваше съкращението ЕИМ (електронно-изчислителна машина). За разлика от останалите изобретени от човека машини, компютрите увеличават не неговата физическа, а интелектуална мощ. Както всяка машина, и компютъра се характеризира с важния за потребителите параметър **производителност**.

Производителността отразява скоростта на обработката на данни (понякога я отъждествяват с бързодействието). Измерва се с брой машинни инструкции или операции, изпълнени за една секунда. Производителите на големите компютри оценяват производителността на апаратната им част с различни мерни единици:

- MIPS- милиони инструкции за секунда;
- MOPS- милиони операции за секунда
- MFLOPS- милиони операции с числа с плаваща запетая за секунда.

За по-големи поръядци на производителността се използват G (гига) или T (тера) операции за секунда.

Много изтъкнати учени имат своя принос в усъвършенстването на най-крупното изобретение на XX век - автоматичния електронен цифров компютър. Истината по авторството на съвременния компютър възтържествува през 1973 година. С решение на Федерален съд на САЩ бе постановено, че **д-р Джон Винсент Атанасов** (от български произход), професор по физика в университета на Айова и неговият асистент **Клифърд Бери** през 1939 построяват първият в света електронен цифров компютър **ABC** (Atanasoff-Berry-Computer) за решаване на системи алгебрични уравнения с голям брой неизвестни. Джон Екърт и Джон Мочли, патентовали през 1945 година цифровата електронно-изчислителна машина ENIAC, некоректно са заимствали устройството и цялата идея за компютъра от тях.

Под **архитектура на компютъра** специалистите разбират тази съвкупност от параметри на компютъра, които са достъпни за програмистите (система от инструкции на компютъра, видовете адресации на паметта, типовете памет, и др.) а под **организация на компютъра** как тези параметри са реализирани (програмно или апаратно, какви са техническите решения за целта). Компютърната архитектура отразява най-общите принципи на взаимодействие на хардуера и софтуера при организацията на изчислителния процес. Тя не разглежда конкретните конструктивни особености на компютъра, управлението и предаването на данните вътре в централния процесор и подобни конкретни въпроси.

Понякога в бита при описание на персоналните компютри под архитектура се разбира състава и структурата на персоналния компютър, функциите , взаимодействието и параметрите на неговите основни устройства. За целта е по-добре да се използва понятието **конфигурация**. Конфигурацията зависи от приложението, необходимите функции и финансовите ограничения пред потребителя.

Терминът **структура** отразява състава, връзките и функциите на апаратните средства на компютъра.

Структурата на хардуера включва четири базови градивни компонента:

- Централен процесор, реализиращ обработката на данни;
- Памет за съхраняване на данните;
- Входно/изходна (В/И) система за обмена на данни между външния свят и централната част на компютъра (процесора и паметта);
- Вътрешни системни връзки, свързващи процесора, паметта и В/И система.

Характерна особеност на компютъра е неговата дискретност - процеса на работа ясно може да се разграничи на отделни стъпки, данните се представят чрез последователност от знаци, принадлежащи на някаква крайна азбука.

В цифровите компютри се използва двоичната бройна система, тъй като техните градивни физически елементи имат само две устойчиви състояния. С едното от тях се кодира двоичната цифра "0", а с другото - "1".

През 1945 г. блестящият математик от унгарски произход **Джон фон Нойман**, работещ в Принстънския университет в САЩ, на базата на постиженията на много изследователи, в това число и на Джон Атанасов, формулира заедно със своя асистент Голдщайн концепцията за последователната архитектура и запомнената програма. Тази концепция включва няколко **архитектурни принципа**:

а) компютърът е електронно устройство и използва двоичен код;

б) структурата на компютъра включва аритметико-логическо устройство, устройство за управление, памет и устройства за въвеждане и извеждане на информация;

в) компютърът работи под управлението на програма (състояща се от инструкции), която се съхранява заедно с данните в една и съща едномерна памет, адресируема чрез последователни номера на запомнящите клетки;

г) инструкциите се изпълняват последователно една след друга, след прочитането им от паметта.

д) докато данните и инструкциите са в паметта, няма явно различие между тях, те се идентифицират само при изпълнение на програмите;

Тези принципи се използват и сега, и поради това се смята, че повече от съвременните последователни компютри са с **класическа архитектура** от фон Нойманов тип. На фиг. 2.1 е представен обобщен модел на компютър с такава архитектура. Там са показани най-главните устройства, характерни за всички компютри.

Централният процесор (ЦП) включва в своя състав аритметико-логическото устройство (АЛУ), управляващо устройство (УУ) и **главна (първична) памет** (тя се състои от **оперативна памет** – ОП и постоянна памет - ПП). Напредъкът на полупроводниковите технологии рязко намали размерите на ЦП. АЛУ извършва всички аритметични и логически операции в компютъра. Това става под управлението на УУ, което дешифрира инструкциите от програмата, запомнена в ОП, и определя последователностите от събития във всички устройства, необходими за изпълнението на тези инструкции. УУ синхронизира всички действия чрез изпращане на електрически импулси. В ОП (английското съкращение е RAM), която е вътрешна за ЦП, се временно се съхраняват програми и данни. За да се изпълнява програма, тя трябва задължително да бъде заредена в ОП. Процесът на въвеждане на данни в ОП се нарича "запис", а извеждането от ОП – "четене". В постоянната памет ПП (ROM), от която може само да се чете, се съдържат специфични програми и данни, свързани с управлението на компютъра (за стартиране, инструкции за операционната система и др.).

Външната памет служи за трайно съхраняване на програми и данни, които не са в ОП (при изключено захранване ОП губи своето съдържание), както и за архивиране на програми и информация.

Периферните (наричат се още "външни") **устройства** са свързани към ЦП чрез специализирани устройства, обединени с термина **интерфейс**. Входните устройства са част от тях, чрез тях се въвеждат данните и се преобразуват в удобна за обработка от ЦП форма. Изходните устройства са периферни устройства, които позволяват компютъра да подава на потребителя информация в разбираема за него форма, или да я предава към други машини.



Фиг.1.1

Развитието на компютърната техника води до непрекъснатото увеличаване на производителността на компютрите, която зависи основно от архитектурата и **технологията** на електронните им компоненти.

От началото на петдесетте години до края на деветдесетте години на миналия век се смениха се **четири поколения цифрови компютри**, различаващи се по електронните си компоненти.

- Първо поколение цифрови компютри (1951 - 1959 г.)-изградени с електронни лампи.

- Второто поколение (1959 - 1969 г.) - изградени с транзистори.

- Третото поколение (1965 - 1971 г.) -с интегрални схеми с ниска степен на интеграция.

- Четвъртото поколение (след 1975 г.) - с интегрални схеми с голяма и свръхголяма степен на интеграция (**ГИС**) и **микропроцесори**. Те позволиха внедряването на микрокомпютри в такива области, където по-рано това бе немислимо.

У нас за интегралните схеми широко се използва названието **чип**.

Започна разработването на компютри от **пето поколение** с елементи на изкуствен интелект.

В зависимост от изпълняваните задачи и стойността им, съвременните цифрови компютри се разделят на:

- а) **суперкомпютри** - най-мощни и най-скъпи (с цена до стотици милиони долари). С тях в напредналите страни са снабдени държавните институции, специалните ведомства, големи научни организации и др. Суперкомпютрите решават много сложни проблеми в режим на реално време (например глобална прогноза за времето, моделиране на ядрен взрив и др.);

- б) **големи компютри** (на запад се използва термина mainframe computers)- за средни и големи компании и научни институти. Могат да обслужват хиляди потребители. Цената им е от няколко стотин хиляди до няколко милиона долара. Класически пример е семейството IBM 360/370, разработено през 60^{-те} и 70^{-те} години;

- в) **мини компютри** - за малки и средни организации и компании. По-малки са от големите, но са достатъчно мощни. Цената им до няколко стотин хиляди долара. Техни представители бяха семействата компютри на американската фирма DEC-PDP 11/LSI 11/VAX II, CM-4, а сега AS 400, използвани за изследователски цели, управление на технологични процеси в автоматизирани заводи, мощни сървъри и др.

г) **микрокомпютри**, с цена до няколко десетки хиляди долара. Дължат появата си на микропроцесора в началото на 70-те години, който предизвика революционни промени в приложението на изчислителната техника. Микрокомпютрите се делят на **универсални** и **специализирани**. Специализираните обикновено се **вграждат** в обекти за управление- автомобили, самолети, електрически и електронни битови прибори и др. **Персоналните компютри** са най-популярните представители на универсалните микрокомпютри. Те са разработени на базата на универсални микропроцесори с възможности за едновременна обработка на 32- и 64- разрядни двоични числа. Непрекъснато се усъвършенстват техните възможности. Американската фирма **IBM** е световен лидер в разработката им със своите персонални компютри **PC**, работещи с микропроцесорите на фирмата **Intel**. Много други фирми произвеждат PC-съвместими персонални компютри, които получиха доминиращо разпространение в света главно заради огромното количество достъпен съвместим с тях софтуер.

Терминът **компютърна система** обединява три съставни части - хардуер, софтуер и човека-потребител, активиращ системата. Според някои автори този термин обхваща и **firmware** (фърмуер)- вграденото в компютъра програмно осигуряване - напр. BIOS- базовата входно-изходна система. Това е по-скоро вид софтуер.

1.2 Представяне на данните в компютрите

Общите категории данни в компютрите са **числа, адреси, символи и логически данни**.

. Числовите данни са три типа: цели числа , реални числа и пакетирани десетични числа.

Бройна система е съвкупност от знакове и правила за записване на числа. В човешката практика се използват два вида бройни системи: непозиционни и позиционни. При **непозиционните** бройни системи теглото на цифрата не зависи от нейната позиция. Често за пример за такава система се дава римската бройна система. 1999 година, написана в римската система е MCMXCIX, а 2007 година – MMVII. В компютрите не се използват непозиционни бройни системи

Позиционна бройна система е тази , в която стойността, която изразява цифрата, зависи от нейното място (**позиция**) в числото.

Десетичната бройна система има десет различни цифри - от 0 до 9, чрез които могат да се представят всички числа. Броя на цифрите в числото определя броя на неговите разряди. Разрядът с най-малко тегло се нарича младши разряд, обикновено той е най-десния в числото. Например за числото 1999 най-дясната цифра 9 има тегло 1, а следващите я отляво надясно съответно 10 и 100.

Хардуерът на компютъра разпознава само два вида електрически импулси, единият от които е с високо ниво на напрежението и обикновено се приема за “1”, а другият, с ниско ниво– “0”. Бройната система, използваща само 0 и 1, се нарича **двоична система**. Единиците и нулите се използват в различни комбинации за представяне на всички числа, букви и символи, които могат да бъдат въведени в компютъра. Двоичната бройна система има за основа числото 2, и само 2 цифри – 1 и 0. Позиционните тегла се базират на степените на основата – 2. Двоичната система не е удобна за ползване от човека и поради това за представянето на данните се използват и други бройни системи – десетична и шестнадесетична.

За да се избегнат грешки, са въведени означения за това, в каква система е дадено число.. Например 11 може да означава 11 в десетичната система, 3 – в двоичната и 17 – в шестнадесетичната. Най-разбираемо означение е изписване на основата на бройната система вдясно като индекс. То затруднява обаче отпечатването при регистрация на данните. За означаване на данните в шестнадесетична система по-

често се използва буквата Н (от Hexadecimal - шестнадесетичен), която се записва непосредствено след числото – 11Н. В програмите това шестнадесетично число се бележат с 0x11. За означаване на число в двоичната бройна система се използва В-например 11В.

За да се преобразува ръчно десетично число в двоично, се използва правилото за деление на цялата част и умножение на дробната част с основата на бройната система, в която трябва да се получи резултатът. Цялата част на изходното число се дели, докато се получи 0. Последният остатък от делението е най-старшият разред на числото в новия код, а първият остатък – най-младия. При преобразуването на дробната част умножението продължава, докато се получи 0 (което не винаги е възможно) или докато се запълнят необходимият брой разреда.

Преобразуването на двоично в десетично число ръчно става с помощта на формула, като най-напред намират разрядите, като този в ляво от запетаята получава номер 0.

Пример: $1100,11В = 1.2^3 + 1.2^2 + 0.2^1 + 0.2^0 + 1.2^{-1} + 1.2^{-2} = 12 + 0,5 + 0,25 = 12,75(10)$

Символът $^$ означава аритм. операция “степенуване”.

За преобразуването могат да се използват и следните съотношения:

$$1_{(10)} = 1_{(2)} = 1.2^0$$

$$2_{(10)} = 10_{(2)} = 1.2^1$$

$$4_{(10)} = 100_{(2)} = 1.2^2$$

$$8_{(10)} = 1000_{(2)} = 1.2^3$$

$$16_{(10)} = 10000_{(2)} = 1.2^4$$

Броят на нулите след единицата е равен на степента на 2.

Шестнадесетичният код се използва главно за съкращаване представянето на многоразредни двоични числа, когато с тях трябва да програмира човек, а не компютър. Основата на тази система е числото 16. То не е избрано произволно – 16 е броят всички комбинации на едно четириразредно двоично число. В шестнадесетичната бройна система се използват цифрите 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, и буквите от латинската азбука А, В, С, D, Е, F, като с буквите се означават десетичните числа 10, 11, 12, 13, 14 и 15. В следващата таблица са посочени двоичните и шестнадесетичните еквиваленти на десетичните числа от 0 до 15:

| <i>Двоично число</i> | <i>Шестнадесетично число</i> |
|----------------------|------------------------------|
| 0000 | 0 |
| 0001 | 1 |
| 0010 | 2 |
| 0011 | 3 |
| 0100 | 4 |
| 0101 | 5 |
| 0110 | 6 |

| | |
|------|---|
| 0111 | 7 |
| 1000 | 8 |
| 1001 | 9 |
| 1010 | A |
| 1011 | B |
| 1100 | C |
| 1101 | D |
| 1110 | E |
| 1111 | F |

Една шестнадесетична цифра се представя с 4 двоични разряда. Шестнадесетичното число лесно може да се преобразува в двоично чрез замяна на всяка шестнадесетична цифра със съответната ѝ група от четири двоични цифри (наречена тетрада) от горната таблица.

$$6B3_{(16)} = 11010110011_{(2)}$$

За да се преобразува число от двоичен в шестнадесетичен код, двоичното число се разделя на групи от по 4 разряда, започвайки отдясно наляво. След това всяка група от 4 разряда (тетрада) се заменя с шестнадесетичния и еквивалент. Ако най-старшата група има по-малко от 4 разряда, тя се допълва с нули отляво (когато числото е цяло, а не дробно).

Науката за законите на мисленето- **логиката**, се занимава и със съжденията-повествователни изречения, за които може да се провери дали изразчват истина или лъжа. Математизирането на тази дейност стана възможно след развиване на математическата логика, наречена още алгебра на логиката или булева алгебра по името на нейния основател Джорж Бул.

За изразяване на логическите съждения се използват **логически функции**, за означаването на отделни съждения- **логически променливи**, а за логическите отношения- специални знаци. За означаване на двете възможни стойности на логическите променливи са достатъчни два знака. Прието е знакът 1 (или **T**- от True) да се използва за истинно съждение, а 0 (или **F**- от False) за лъжливо. Това дава възможност съжденията да се разглеждат като двоични променливи, а логическите отношения- като функции с аргументи двоични променливи.

Всяка логическа функция може да се изрази чрез елементарните логически функции **конюнкция** (логическо умножение “И”- означава се и със знака &), **дизюнкция** (логическо събиране- “ИЛИ”) и **отрицание** (логическо “НЕ”). Често се използва и логическата функция **изключващо или**, наричана сума по модул 2 или логическа неравнозначност. Връзката между входните променливи (аргументи) и изходните величини на една логическа функция може да се изрази чрез **таблица на истинност**. Таблицата на истинността се състои от две части. Първата част съдържа всички възможни състояния на входните величини, а втората част определя изходните състояния на логическата схема като функция на съответната комбинация от състоянията на входните величини.

Например таблицата на истинност за логическата функция И- **Z** от две логически променливи **a** и **b** е следната:

| z = a & b | a | b |
|--------------------------|----------|----------|
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 1 | 1 |

Логическите функции се използват при проектирането на основните устройства на хардуера на компютрите, при реализирането на логическите операции в работата на АЛУ и при програмирането.

Освен числа, компютрите трябва да обработват редица символи, като @, #, * и букви. Тези символи и букви, въведени например от клавиатурата, трябва да се преобразуват в двоичен вид преди да се обработят от компютъра. За това преобразуване са разработени кодиращи системи, най-популярната от които е **ASCII**. Чрез този седмразряден код могат да се представят 128 различни символи- всички главни и малки букви от латинската азбука, цифри, знаци за пунктуация, символи за аритметични действия и разни символи за управление. Значенията на някои букви от латинската азбука, цифри и символи, изразени в шестнадесетичен код са следните:

A = 41; a = 61; @ = 40;
 B = 42; b = 62; 1 = 31; 9 = 39.

В паметта кода на буквата А се записва в двоичен вид 100 0001.

Тази система за кодиране е международен стандарт. Ако една програма работи с текст, съставен от различни символи, всеки от тях се кодира чрез един байт. За представяне на числа чрез ASCII - код, за всяка цифра е необходим 1 байт. Например числото 57₍₁₀₎ ще бъде записано в два байта като 3537H (буквата H означава, че числото 3537 е шестнадесетично). Такова представяне се нарича още "непакетирано" или "разпакетирано". Трябва да се помни, че числата изведени на екрана на видеомонитора или на печат, трябва да бъдат представени в ASCII - код.

През последните години все по-голямо разпространение получава кодиращата система **UNICODE**, в която за всеки буквен или друг символ се отделят по 2 байта.

Единственото средство за представяне на данните са електронните елементи, всеки от които в даден момент от време се намира в едно от двете си възможни състояния. Това състояние условно изразява стойността на една двоична цифра – "0" или "1". Една двоична цифра е един разряд от многоразредно двоично число. Всеки елемент, представящ един двоичен разряд се нарича разрядна клетка. Разрядната клетка представя елементарната информационна единица "бит" (означава се с "b"). Многоразрядните двоични числа се записват чрез поредица от последователни разрядни клетки, образуващи абстрактното поле "разрядна мрежа".

Чрез разрядната мрежа целите числа се представят във формат с фиксирана запетая, а реалните- с плаваща запетая.

Представянето на двоичните числа във формат с плаваща запетая се използва за апроксимиране на реалните числа.

Число X във формат с плаваща запетая може да се представи в общ вид така:

$$X = zM \cdot 2^{zP}$$

Тук M е мантиката на числото X, което е положителна или отрицателна правилна дроб (т.е. |M|<1), p е порядъка (среща се и термина експонента), z е знака "плюс" или "минус". Диапазонът на представяните числа с плаваща запетая е значително по-широк от този на представяне с фиксирана запетая.

Много от водещите фирми в света, между които е Intel, възприеха стандартите на IEEE 754 за форматите за представяне на данните с фиксирана запетая и плаваща запетая - **дума**- 2 байта, **двойна дума**- 4 байта, **четворна дума**- 8 байта.

Използва се и двоично-десетичното представяне на числата (BCD). Те могат да бъдат непакетирани, когато 1 – BCD цифра се представя в 1 байт (с 8 бита) и пакетирани, когато 2 цифри се представят в 1 байт. За кодирането на всяка десетична цифра се използват двоичните числа от 0000 до 1001.

Компилаторите на езиците за програмиране от високо ниво заделят необходимото място за числата в паметта на компютъра, изхождайки от зададения тип на данните. Например ако в един програмен ред от програма на езика Java е написано `short age;`, това означава деклариране на целочислената променлива `age`. Компилаторът след интерпретирането на този оператор ще задели в оперативната памет два байта за тази променлива, в които може да се записват различните ѝ стойности, представени с фиксирана запетая.

В АЛУ на компютрите има само сумиращи устройства, поради това се налага аритметичната операция "изваждане" да се сведе до сумиране на отрицателни числа. Следователно сновната аритметична операция в компютрите е събирането. При това се налага знака на всяко число да се отрази по някакъв начин в двоичния му код. Обикновено към числото се добавя допълнителен разряд, наричан знаков. Този разряд се разполага пред най-старшата (най-лявата) значеща цифра на цялата част на числото. Прието е положителният знак (+) на числото да се означава с 0, а отрицателният (-) – с 1. Най-често използваните двоични кодове за представяне на числата със знак в микрокомпютрите са правия, обратния и допълнителния. Чрез допълнителният код в компютрите е удобно да се извършват операциите алгебрично сумиране (свежда се до аритметичното събиране на техните кодове) и изваждане (то лесно се свежда до алгебрично събиране : $a - b = a + (-b)$). При събирането на две числа са в сила зависимостите: $0 + 0 = 0$, $0 + 1 = 1$ и $1 + 1 = 0$ (получава се и пренос 1 към старшия разряд). Всички останали операции – изваждане, умножение, деление, повдигане на степен, тригонометрични функции, и т.н. могат да се сведат до сумиране на двоични числа. Разработените алгоритми за изпълнение на аритметичните операции се базират на сложни числени методи.

1.3 Структура на персонален компютър

Стационарните персонални компютри, произведени от фирмата IBM или съвместими с тях, независимо от техните технически характеристики, са популярни с английското съкращение "PC".

Персоналният компютър съдържа следните основни елементи, които са необходими за изпълнение на всяка програма:

- микропроцесор;
- основна памет;
- шина;
- контролери за въвеждане / извеждане (В/И);
- твърд диск и други външни запомнящи устройства (флопидисково устройство, CD-ROM);
- видеомонитор (дисплей);
- клавиатура и мишка, за взаимодействие с програмите и въвеждане на данни;
- периферни устройства за извеждане на резултатите (принтер, плотер).

Тези компоненти са свързани чрез съвокупност от проводници- системната шина (магистрала). На фиг.1.2 е показана структурата на персонален компютър, към системната шина на когото са свързани процесор с външната му буферна (кеш) памет, основната памет, клавиатура и външно запомнящо устройство с твърд магнитен диск

(означено е с HDD) чрез специални входно-изходни устройства - контролери (управляващи устройства), както и видеокарта - модул за управление на видеомонитора. В ранните модели персонални компютри към процесорите се подаваха тактови импулси от обособен външен тактов генератор със стабилизирана честота за задвижване и синхронизиране работата на компютърните елементи. Интервалите между отделните импулси определят така наречения машинен такт. В модерните компютри този генератор конструктивно е интегриран с останалите интегрални схеми.



Фиг.1.2

Компютърната шина се състои от много паралелни проводници, всеки от които в даден момент носи един бит (част от адрес, инструкция или данни). Обособена група от тези проводници предава синхронизиращи импулси, друга - управляващи сигнали, трета- двоичен код, който идентифицира адреса на компонента (процесор, памет, периферен контролер) работещ с данните (нарича се адресна шина- **АШ**), а друга група проводници предава сигнали, кодиращи данните (шина за данни- **ШД**).

Модерният персонален микрокомпютър притежава няколко различни области, в които се използват шини. Едни са шините в така наречената "микропроцесорна област", обслужващи процесора и паметта, като ускореният графичен порт за видеокарти **AGP**. Друга е шината, обслужваща високоскоростните входно-изходни устройства.

Стандартните компютърни шини са **ISA**, **PCI** и **USB**.

Интерфейсът е система от шини, електронни компоненти към тях, свързващи кабели, сигнали и алгоритми и конвенции, реализирани в схемите за управлението му, предназначена за стандартизирано свързване на компютърните компоненти и управление на обмена на информация между тях. Той се използва за включването към компютъра на печатащи устройства, модеми и др. В персоналните компютри интерфейсът се представя от паралелния и серийните портове.

Паралелния порт(LPT) реализира едновременното, паралелно предаване на няколко бита, по отделен проводник за всеки бит (обикновено Byte). Той традиционно се възприема като порт за свързване на печатащо устройство към компютъра. **Последователният порт (COM1,COM2)** осигурява предаването на данни по една линия последователно бит след бит. Други синоними за него са комуникационен порт, RS-232 порт, порт за асинхронна комуникация на данни. Към него се свързват модеми, мишки и др. устройства.

Най-общата организация на работата на компютъра е следната.

Веднага след включването си, компютърът РС изпълнява най-напред програмата за начално самотестване POST (Power - on - self – test). Когато тази програма открие неизправност в някой от апаратните компоненти, тя извежда съответните съобщения върху екрана или издава поредица от звукови сигнали ("писукания").

Еднократният звук и извеждането на екрана на ОС-маркер е признак за изправност на всички компоненти. След проверката с програмата POST компютърът е готов за зареждане на операционната система от диска.

Чрез програма, записана в постоянната памет, автоматично се стартира процедура за начално зареждане на операционната система.

Процесорът получава от основната оперативна памет инструкции за своята работа (те са част от неговата програма). След това от външния диск или от оперативната памет той извлича нови данни, обработва ги и ако е нужно ги връща обратно в паметта или ги изобразява на видеомонитора. След включването на компютъра микропроцесора изпълнява този основен цикъл десетки милиони пъти в секунда. Всяка стъпка, на която се извлича инструкция или данни и в паметта се изпращат резултати, води до предаване на информация по шината и възможно, по контролерите за В/И.

Конструктивно персоналният компютър се състои от **системен блок**, поместен във вертикален или хоризонтален корпус, и редица периферни устройства, свързани чрез кабели със системния блок.

В системният блок са монтирани:

а) основна платка (така нареченото "дъно" на компютъра), свързваща в едно големите интегрални схеми на микропроцесора, основната оперативна памет и другите електронни устройства на компютъра;

б) захранващ блок на компютъра, осигуряващ преобразуването на входното мрежово напрежение от 220 V във всички необходими за частите на компютъра напрежения;

в) външни запомнящи устройства (твърд диск, флопидисково устройство, CD-устройство и др.);

г) допълнителни контролери (специални електронни схеми за връзка с различни периферни устройства) и др.

Дънната платка (**дъно**, motherboard) е един от най-важните компоненти на РС. Тя представлява печатна платка, върху която е събрана по-голямата част от основната електроника- съдържа чипове CPU, RAM, ROM- BIOS, Chipset, слотове за разширение, входно- изходни портове и др. От тази платка много зависи надеждността, бързодействието и възможността за разширение и модернизация.

Процесорът лесно се открива върху дънната платка, тъй като е с най-големи размери, има охлаждащ вентилатор. Могат да се открият надписи, съдържащи фирмата – производител- Intel, AMD, или CYRIX.

Чипсетът (схемният набор) е важен компонент на дъното. От него зависи връзката между електронните компоненти. Това е комплект от чипове, който включва контролер на паметта, кеш контролер, контролери на В/И устройства, IRQ и DMA контролери, контролери за външни запомнящи устройства и др. Чипсетът определя типа на цокъла на ЦП, типа и макс. обем на ОП и др.

Най-разпространената архитектура на чипсета е със северен и южен мост.

Северният мост отговаря за трансфера на данни между ЦП, основната системна памет и порта AGP. **Южният мост** следи за В/И операции, в това число поддръжката на шината USB, на серийния и паралелния порт и на такъв за клавиатура, на контролери с интерфейс IDE за твърди и флопи дискове. Тези мостове общуват помежду си чрез системната шина PCI.

Видеомониторът на компютъра служи за изобразяване на екрана на графична и буквено-цифрова информация за потребителя. Персоналните компютри използват видеомонитори, които по своя принцип на работа не се отличават от тези на цветните телевизори. Освен видеомонитори с електронно-лъчеви тръби, за мобилните и стационарните персонални компютри са разработени плоски дисплеи с различни принципи на формиране на изображението (течни кристали, газово-плазмени, електролуминисцентни, светодиодни, и др). Технологията с течни кристали (**LCD**) е втората по разпространение след ел.лъчевата тръба технологии, но тя е за **плоски** видеомонитори. Означава се с **TFT**. Типични параметри за такива монитори за стационарни компютри са- **диагонал на екрана**- 17"- 19" (инча); **макс.резолюция**- 1280 x1024 пиксела; **яркост**- 270-300 кандела / кв.м; **контраст**: 500:1 - 2000:1 (степени на сивото); **големина на пиксела**: 0.264-0.294 мм (съставна цветна точка на екрана);**време на реакция** : 2-12 ms (скорост на промяна на цвета на точката); сертификати: TCO- 02 , - 03 ,TCO 99- (степен на излъчване).Допълнителни характеристики- вградени тонколонки, телевизионни тунери.

Клавиатурата е основно периферно устройство за въвеждане на букви, цифри и символи в компютъра - входни данни и команди за неговото управление. Тя се свързва чрез контролер на клавиатурата с шината на компютъра. При всяко натискане на клавиш от електронните схеми в клавиатурата се генерира позиционен код, а контролера го преобразува в ASCII-код. Този код по-нататък се изпраща в паметта и се използва при изобразяване на този символ на екрана на видеомонитора или отпечатване с принтер на съответната позиция, зададена от програмата.

С широкото прилагане на графичния потребителски интерфейс при всички съвременни програмни продукти, задължителен архитектурен елемент в съвременния микрокомпютър стана периферното устройство **мишка** . Използването на устройства за въвеждане като **джойстик**, мишка или **трекбол** подобрява значително възможностите за обратна връзка между компютъра и неговата периферия. Чрез тях се управлява дисплейния маркер върху екрана.

За регистриране на резултатите от обработката на данни с компютъра се използват печатащите устройства (**принтерите**). Съществува голямо разнообразие от принтери, отличаващи се по способа за формиране на символите върху хартията, скоростта си на работа, цена и др. Най-напред бяха внедрени мозаечните принтери, в които всеки символ се отпечатва последователно чрез удари от подвижна печатаща глава с набор от 24 игли. Предимството им за едновременно отпечатване на повече от едно копие и изключително евтините консумативи все още ги задържа на пазара.

Сега се използват предимно мастилено-струйни и лазерни принтери.

Най-важните характеристики на цветния мастилено- струен принтер, са: гнезда за касети с мастило-2; резолюция на печата- 9600 x 2400 точки на инч (dpi);скорост на отпечатване- до 29 страници/ мин, формат А4. Свързване с компютър- чрез шина USB .

За черно-бял лазерен принтер, освен марката и цената, са характерни скорост на отпечатване- примерно 20 стр. (формат А4) за минута ; резолюция- 1200x1200 точки на инч (dpi); отпечатани страници А4 за месец- 15 000 (нарича се цикъл);памет 32 MB. Свързване: с USB и паралелен порт

Има комбинирани **мултифункционални устройства**- принтер, скенер, факс и копирна машина в един корпус.

За автоматичното въвеждане на графична информация (снимки, чертежи и др.) в компютрите се използват **скенерите**. Типични техни характеристики са резолюция на сканиране : 3200 x 6400 dpi (точки на инч); скорост на сканиране: 33 секунди(1 лист формат А4); дълбочина на цветовете- 48 бита (по 8 бита за всеки основен цвят).Свързване -: чрез USB

За извеждане на графична информация от системите за автоматизирано проектиране (т.н. CAD-системи) се прилагат графични регистриращи устройства (**плотери**), които изчертават със специални пера точни графични изображения, схеми и други на проектираните изделия. Звуковата карта (платка) се отнася към мултимедийните архитектурни компоненти на съвременния микрокомпютър. Към нея се включват стереозвукови колони.

В мултимедийните системи и системите за виртуална реалност има възможност за включване на периферни устройства - видеокамери, цифрови фотоапарати, тонколони, шлемове за виртуална реалност, ръкавици с тактилни датчици и др. Компютърна Web-камера има следните характеристики: макс.резолюция- 640 x 480 пиксела, свързване – USB, възможно вграден микрофон.

1.4 Състав и функции на процесора

Централният процесор (ЦП) е предназначен за управление на всички устройства на компютъра и изпълнение на операциите за обработка на данните.

ЦП на един модерен персонален компютър физически представлява миниатюрен силициев "чип" (това е жаргонен термин за интегрална схема) и се нарича "микропроцесор". В него има милиони транзистори и връзки между тях, и те формират електронните вериги на централния процесор. Макар че процесорът има много малки размери, той се състои от редица функционални блокове и устройства.

Аритметико-логическо устройство (**АЛУ**) съдържа електрически схеми, които обработват данните.

В централният процесор се съхраняват малко на брой данни в няколко десетки регистра. **Регистърът** е устройство в микропроцесора, което може да запомни само една двоична дума с определена дължина, но е изключително бързодействащо . Данните се извличат от оперативната памет и временно се съхраняват в тези регистри на процесора. Само след като тези данни са обработени от веригите на АЛУ, междинните резултати от това се изпращат отново в регистрите. Крайните резултати се изпращат в оперативната памет.

Управляващото устройство (**УУ**) е централно за процесора, то има синхронизиращи и управляващи вериги. Това устройство определя основните фази на изпълнение на една инструкция (процесорен цикъл) -"извличане – декодиране – изпълнение". Освен управлението на цикъла на изпълнение на инструкциите, управляващото устройство декодира инструкциите и адресите (намира разположението на исканите данни). Управляващото устройство има регистри за управляваща информация. Това са програмният брояч (ПБ), регистъра на инструкциите (РИ) и флаговия регистър (означава се с F). Програмният брояч съхранява адреса на клетки от паметта, където се намира следващата за изпълнение инструкция. РИ съдържа поредица от битове, представящи текущата инструкция – отделни полета на тази битова поредица представят кода на инструкцията и адресите на исканите данни.

Повечето ЦП имат така наречения **флагов регистър**. Отделните битове от този регистър съдържат различни данни за състояния в процесора.

Микропроцесорът изпълнява поредица от инструкции, от които е съставена машинната програма. Програмата може да бъде представена като последователност от инструкции в паметта. Всяка инструкция определя или стъпка на компютъра за обработка на данни, или управляващо действие. Тя задава ясно и точно действието на процесора, без да оставя нищо неопределено. В инструкцията има информация за операцията, която трябва да извърши компютъра и за това кои операнди (стойности) ще

участват в тази операция. В двоичното представяне на инструкцията те формират поле за код на операциите и адресна част.

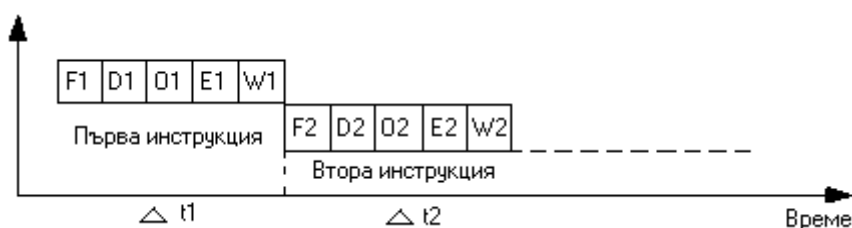
Работата по изпълнение на програмата стартира със запис в програмния брояч на адреса на клетката от паметта, съдържаща първата инструкция от програмата. След това започва цикъл "извличане – декодиране – изпълнение". ЦП изпраща искане за извличане от паметта по адрес, специфициран от програмния брояч ПБ, след това той приема поредната инструкция и я съхранява в регистъра на инструкциите РИ. След това веднага се обновява съдържанието на ПБ с адреса на следващата за изпълнение инструкция. Инструкцията от РИ се декодира и изпълнява.

При обработката на всяка от инструкциите на машинната програма от микропроцесор могат да се разгледат следните основни фази:

- 1) Извличане на инструкцията от паметта по зададен адрес в програмния брояч (ПБ) и изпращането и в регистъра на инструкциите (РИ) - тази фаза се означава с F.
- 2) Декодиране от управляващата логика на микропроцесора на типа и предназначението на извлечената инструкция. Съдържанието на програмния брояч (ПБ) се променя така, че да показва адреса на следващата за изпълнение инструкция - фаза D.
- 3) Ако инструкцията използва данни от паметта, се определя адреса на операндите и те се извличат от паметта в някои от регистрите с общо предназначение на микропроцесора - фаза O.
- 4) Изпълнение на инструкцията - фаза E.
- 5) Съхраняване (запис) на резултата в определено място (в паметта или регистър на процесора) - фаза W.

Следващият цикъл на изпълнение на следващата инструкция започва отново от пункт 1.

В микропроцесорите от първа генерация (това са 8-битови микропроцесори от средата на 70-години на XX век), в които има един универсален блок за изпълнението на всички фази на инструкциите, това става последователно във времето съгласно схемата, посочена на Фиг.2.3.



Фиг.1.3

Тази схема на изпълнение не е икономична по отношение на процесорно време. В микропроцесорите след втора генерация, чрез въвеждане на конвейерната обработка, се постига паралелно изпълнение на няколко инструкции, но за целта в структурата на тези процесори са въведени специализирани блокове за изпълнение на отделните фази от цикъла на инструкциите.

Основни производители на микропроцесори и техни продукти са **INTEL**- Pentium IV, Itanium, Pentium M, Pentium D, **AMD**- Sempron, Athlon 64 FX, Opteron, **Cyrix**- MXi, МП, **IBM/ Motorola**- PowerPC 603,- 750 (G3) и **Digital/Compaq**- Alpha.

Основните характеристики на един микропроцесор са:

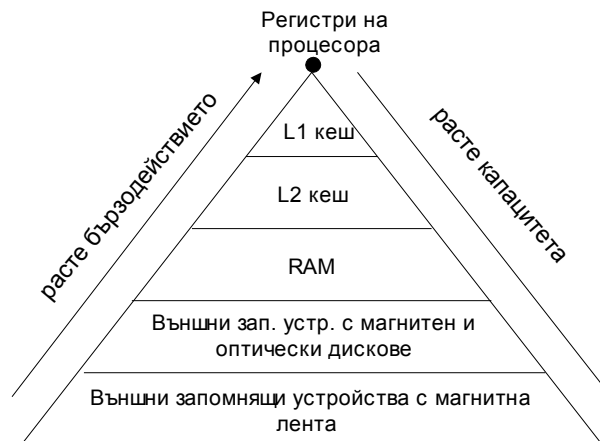
- Тип (напр. Pentium Duo, Athlon XP);
- Ширина на шината за данни (от там следва разрядност на процесора);
- Бърздействие (обикновено чрез тактовата честота);
- Кеш- памет –тип и капацитет;
- Групи процесорни инструкции, които поддържа (напр.SSE, MMX, 3D Now и др.);
- Тип на корпуса и съединителя.

1.5 Компютърна памет

За потребителят на персоналния компютър е по-важно да знае с какъв обем оперативна и външна памет разполага, а не толкова какво е физическото ѝ разпределение по видове. Цялата памет на компютъра може да се представи като йерархична пирамида, на чийто връх са регистрите на процесора и неговата кеш-памет, а на по-ниско ниво са външната за процесора кеш-памет, оперативната памет и външните запомнящи устройства (фиг.2.4). Регистърът е устройство в микропроцесора, което може да запомни само една двоична дума с определена дължина, но е изключително бързодействащо.

Основните характеристики на компютърната памет са **капацитет**, **бърздействие** (време за достъп) и **скорост на предаване на данните** към компютъра.

В основата на пирамидата на фиг.1.4 са външните запомнящи устройства на магнитни дискове, CD, DVD, ленти и др. Последните имат най-голям капацитет, но най-малко бърздействие, докато при изкачване към върха на пирамидата намалява капацитета, а се увеличава бързодействието на видовете памет .



Фиг. 1.4

Данните се прехвърлят от едно физическо ниво към друго на блокове, под управление на операционната система. Ненужните в момента масиви от данни се връщат във външната памет, а необходимите се записват от външната в оперативната памет, а от там в кеш-паметта към процесорните регистри.

За да може да работи ефективно една компютърна конфигурация, то тя трябва да притежава достатъчна по количество оперативна памет.

Оперативната памет съхранява инструкциите и данните, използвани от процесора, в двоичен код. Тя е съвкупност от клетки, всяка от които "помни" двоично число. Обикновено в персоналните компютри всяка клетка е с дължина 8 бита, или това е 1 байт. Оперативната памет е адресируема (информация от нея се извлича след посочване на адрес), микропроцесорът обработва едновременно няколко байта, наречени машинна дума. ОП е изработена по полупроводникова интегрална технология от различен тип и

с различни дебелини на шините в тях (в микрони). Най-масово използвания тип оперативна памет в персоналните компютри е динамичната памет DRAM (Dynamic Random Acces Memory). По-стари типове ОП са **EDO** и **FPM**, а съвременни са **SDRAM**, **DDR**, **DDR2** и **Rambus DRAM**.

При запис на информация в динамичната оперативна памет, софтуерът, съвместно с операционната система, изпраща електрически импулси по избраната от местното управление адресна шина. Тези импулси указват къде сред множеството клетки в RAM-чипа трябва да се запишат данните. Запомнящите клетки са изградени чрез миниатюрни кондензатори. Тъй като кондензаторите, които съхраняват данните в RAM паметта, бавно се разреждат (за няколко милисекунди), е необходимо техният заряд да бъде периодично опресняван. За това се грижи специална част от схемата за управление на паметта, която през определени интервали от време възстановява зарядите на кондензаторите в чиповете. Когато захранването на компютъра е изключено, всички кондензатори изгубват заряда си.

При работа на съвременните бързи микрокомпютри се стига до ситуация, когато микропроцесорът трябва да преустанови своята работа, за да изчака получаването на данните от паметта. Използването на обикновена памет от тип DRAM за съхранението на данните води до голямо несъответствие между бързодействието на микропроцесора и паметта.

Решението на проблема се постига с използването на бързодействаща буферна памет, наречена кеш-памет. Тя и контролера за управлението ѝ са посредник между микропроцесора и основната памет. Конструктивно кеш-паметта може да бъде вградена вътре в самия микропроцесор (кеш от първо ниво, Level -1 кеш или L1) или да бъде изнесена извън него (кеш от второ ниво, Level - 2 кеш или L2).

Кеш-паметта се изгражда чрез статични полупроводникови паметни **SRAM** (Static Random Acces Memory) с време за достъп до елементарна запомняща клетка, междинно в сравнение с това на микропроцесора и "бавната" основна DRAM. Статичната памет има такава висока скорост на работа поради липсата на цикъл за възстановяване на данните, както при динамичната памет.

Предаването на информацията от паметта до процесора и обратно, както и между останалите елементи става чрез **шината**. Предаването се извършва на порции от по няколко байта едновременно. Съществува специална организация на взаимодействие между оперативната и външната памет по такъв начин, че потребителят сякаш разполага с много по-голям капацитет от физически достъпната. Тя се нарича **виртуална памет**. Управлението и е основно чрез операционната система на компютъра.

В компютрите има и постоянна памет (**ROM**), от която само се чете, и тя не губи съдържанието си при изключване на захранването. В нея се съхраняват важни данни и системни програми, наречени общо **BIOS**. Чрез тях се стартира компютъра и се изпълняват входно-изходни операции. В съвременните системи се монтира Flash ROM, която може да се препрограмира електрически без да се вади от компютъра, по този начин се актуализира BIOS с нови версии. По същата технология са разработени и популярните мобилни миниатюрни **флаш-памет** за пренос на програми, музика, изображения и др. Техния капацитет е вече няколко GB, скоростта на трансфера на данни е около 15 Mbs. Трябва обаче да се помни, че броя на презаписите върху тях не е безкраен.

Веднага след включването на компютъра неговата оперативна памет е като чист хартиен лист и обратно – след неговото изключване всичко, което е написано в нея, като правило изчезва (повечето ОП не помнят данни без електрическо захранване). За да не става това, компютрите имат **външни запомнящи устройства (ВЗУ)**. Те се

използват за дълговременно съхраняване на програми и данни и за архивиране на информация. Средата, в която се запомнят данните, се нарича информационен носител. Според принципа си на действие, най-популярните за персоналните компютри ВЗУ се разделят на три класа:

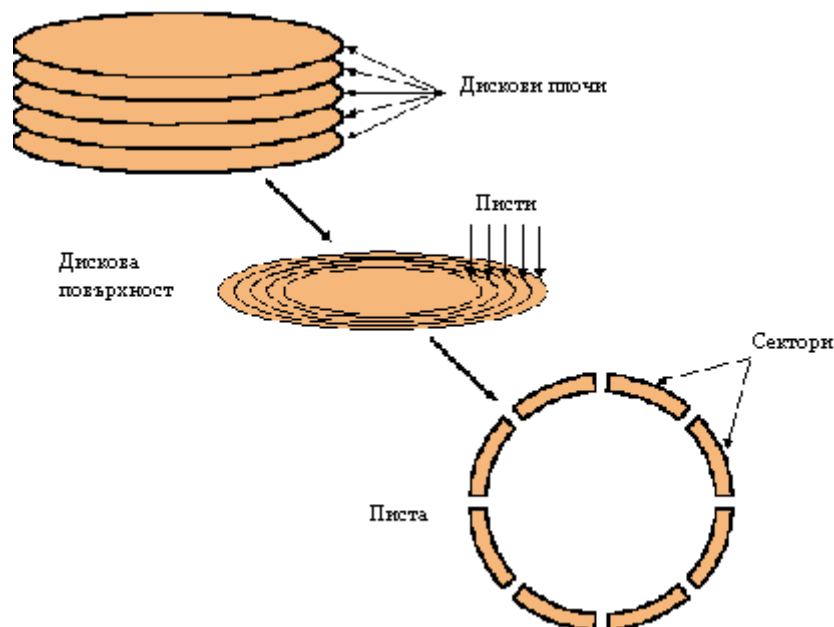
- а) оптически;
- б) магнито-оптически;
- в) магнитни.

В тези ВЗУ се използват основно дискови информационни носители-оптически дискове, сменяеми магнитни дискове във флопидисковите устройства и несменяеми магнитни дискови пакети (твърди дискове, hard discs).

Оптическите ВЗУ използват лазерна технология за запис и четене на информация върху оптически компакт диск (CD, DVD). В тях записът на информация е цифров. Капацитетът на оптичен диск е от 650 MB до около 5 GB, и това не е предел. Съществуват няколко разновидности на тези устройства в зависимост от възможността само да се чете, или да се записва и чете информацията от оптическия диск.

Магнитните ВЗУ са най-разпространените външни памети за персоналните компютри. В зависимост от времето за достъп до блоковете от информация, те се разделят на ВЗУ с пряк достъп – дискови запомнящи устройства, и ВЗУ с последователен достъп – лентови запомнящи устройства (стримери).

Съхраняването на данни върху магнитните носители се базира на цифровия магнитен запис. За целта трябва да взаимодействат два елемента - информационен носител (диск, лента), върху чиято повърхност е нанесен магнитен материал, и магнитна глава, с която се извършва записа или четенето на информация върху/от носителя. Информацията се разделя на логически блокове, като всеки блок се записва в един **сектор**. Броят на секторите в една **писта** е различен (фиг.2.5). Секторът е най-малката адресируема единица във ВЗУ. ОС може да определя размера на блоковете от 512B до 4096B (понякога и повече).



Фиг. 1.5

Преди съхраняване на данните върху диск, той трябва да бъде форматиран. Това форматиране създава нещо като “пътен указател”, който показва на дисковото устройство къде да съхранява и къде да намира данните.

Дисковият контролер е основен елемент на дисковата подсистема . Той е предназначен да управлява ВЗУ и да прехвърля данните между диска и системната шина на компютъра. Контролерът има **дискова кеш-памет**. Това е “скрита” памет, където обикновено по-често използваните данни се пазят по по-бърз достъп. Тя осигурява паралелната работа на процесора по обмена на данни с диска и изпълнението на друга задача. Дисковата кеш-памет понякога работи като кеш-паметта на процесора. Ако в нея има всички необходими данни за процесора, той не губи време за обръщане към диска. Основната разлика между процесорната кеш-памет и дисковата е в използването им. Първата е буфер между устройства с различно бързодействие, а втората намалява рязко интензивността на използването на бавния диск.

Цифрите, буквите и другите знаци, записани във външните запомнящи устройства са данни. **Файлът** е именувана последователност от данни. Той може да бъде разположен както във ВЗУ, така и в ОП.

Основните параметри на хард дисковите устройства за персоналните компютри са **капацитет**, скорост на въртене на диска, **време за достъп** до сектор, скорост на трансфер на данните и типа на интерфейса към дисковия контролер. Има външни дискови устройства с параметри: размери на цигарена кутия, тегло около 200 г ; Plug&Play инсталация; капацитет – 60 GB, скорост на въртене 5400 rpm, дисков кеш- 2 MB; скорост на пренос на данни 480 Mbps; свързване- USB 2.0.

1.6 Входно изходна система

Независимо от класа, към който принадлежи, всеки компютър може да се разглежда като съвкупност от две части - централно ядро (в него влизат ЦП и оперативната памет) и средства за връзка с външния свят- периферни (външни) устройства.

Входно-изходната система е съвкупност от апаратни и програмни средства, предназначена за обмен на информация между централната част на компютъра и външната среда. С развитието на компютърните системи се променяше и съдържанието на този термин. Най- общо системата съдържа канали, интерфейс и периферни устройства. Най-разпространеният вариант е система с интерфейс тип "обща шина", която е характерна за съвременните мини и микрокомпютри.

Включването на периферните устройства към системната шина на процесора става чрез интерфейса за вход/изход. **Интерфейсът** осигурява стандартизирано свързване към компютъра на устройства от различни производители. Системният интерфейс на компютъра е различен от интерфейса на периферните устройства. В персоналните компютри се използва стандартния интерфейс RS-232C за свързване на мишки, модеми, плотери и др. към последователните портове, а за свързване на принтери- паралелен интерфейс.

Порт от програмна гледна точка е група от двоични разреди, до които процесорът има програмен достъп при входно/ изходна операция. Логически той може да се приеме като буферен регистър. Всяко периферно устройство е съвкупност от няколко порта, която се нарича програмен модел на това устройство.

Входно/изходната система на персоналните компютри има два основни режима на работа - програмно-управляем обмен на данни (ПУОД) и апаратно управляем обмен, наричан още пряк достъп до паметта (ПДП, в литературата се среща често и еквивалентното му английско съкращение DMA).

ПУОД има няколко разновидности, най-използвания в съвременните персонални компютри е програмно- управляемия обмен чрез **прекъсване** на основната програма на процесора. Той се осъществява чрез сложно взаимодействие на специализирани електронни схеми и програми от операционната система. Когато периферното устройство е готово за въвеждане или извеждане на данни, то изпраща сигнал, наречен "**Заявка за прекъсване**", към процесора. ЦП спира изпълнението на основната програма на компютъра, автоматично запомня текущото състояние на важните системни регистри в област от оперативната памет, наречена "**стек**" (тя работи на принципа **LIFO** - "последен влязъл - пръв излязъл"), и прави преход към специална програма за обработка на прекъсването (наричана още "обслужваща програма", тъй като тя управлява процесора по обслужване на заявката) . След като програмата за обслужване на прекъсването приключи своята работа, автоматично се възстановява "старата" системна информация, прочетена от стека и процесора продължава изпълнението на основната си програма от мястото, в което е бил прекъснат. За всеки източник на прекъсване в паметта на компютъра има обслужваща програма .

За работа с бързи периферни устройства (външни запомнящи устройства), се използва прекия достъп до паметта. ПДП се осъществява от електронните схеми в компютъра, които директно предават данните от буфера за данни на периферното устройство към оперативната памет.

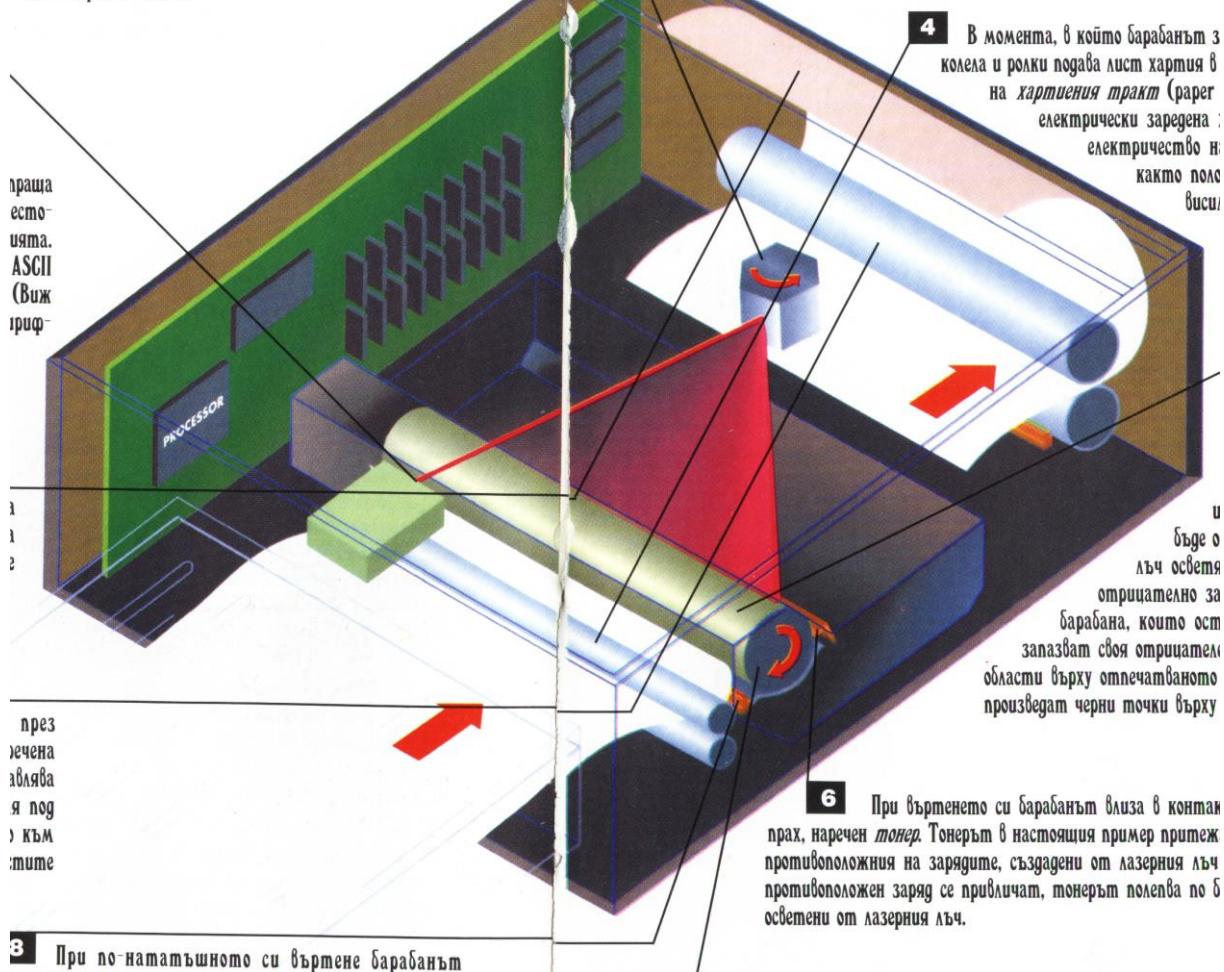
Свързването на периферните устройства в персоналните компютри става основно към системната шина. В по-старите компютри това бе 16- битовата шина ISA със скорост на предаването на данни около 5 Mb/s , а в съвременните- 32 или 64 битовата PCI със скорост около 132 Mb/s и 64-битовата AGP със скорост над 528 Mb/s (за включване на графични устройства).

Периферните устройства се разделят на две важни групи. В първата влизат устройства, които осигуряват съхраняване на данните - това са външните запомнящи устройства..Втората група включва периферни устройства, които свързват компютърната система с външния свят. Те се наричат входно/ изходни устройства, например клавиатури,принтери, видеомонитори, датчици и др. Има разработени голям брой периферни устройства, които се различават по функция, партньор (с кого взаимодейства) и скорост на предаване на данните. Всяко периферно устройство има устройства за управление (контролери). Те изпълняват функции по управление и синхронизация, комуникации с ЦП и устройства, буферирание на данни и откриване на грешки.

Схема на лазерен принтер

2 Процесорът на принтера управлява лазерен лъч като го пропуска или го гаси посредством бърз електрически превключвател.

той се завърта - обикновено на 1/300 част от инча при повечето лазерни принтери обхожда следващия ред от точки.



3 При по-нататъшното си въртене барабанът

1.7 За паралелните компютърни архитектури

Върху компютърните архитектури оказват въздействие технологията на развитие на елементната база на компютрите, езиците за програмиране, операционните системи, приложния софтуер. При разработване на нова компютърна архитектура се подбира най-напред **системата от инструкции**, а тя зависи от областта на приложение на бъдещия компютър. Системата от инструкции е списък от команди на ниво машинен код, които може да реализира конкретния процесор. Тя характеризира изчислителните способности на компютъра на най-ниско апаратно ниво. Всяка система от инструкции е машинно ориентирана.

Компютрите, изградени по класическа архитектура, работят последователно във времето и това не е особено ефективно и рационално от гледна точка на работата на цялата система. Традиционната фон Нойманова архитектура не е достатъчно ефективна при решаване на задачи с голям обем изчисления и работа в реално време. Едно от перспективните направления за повишаване на производителността на компютрите е **паралелната обработка**. Тя използва множество процесори, които обработват едновременно големи масиви от данни.

Компютърът, извършващ паралелна обработка, обработва в един и същ момент от време няколко инструкции от една програма чрез няколко централни процесора. Има разлика между паралелната обработка и работата на **многопроцесорните системи**. В последните няколко програми се изпълняват едновременно, но всяка на различен процесор. **Суперкомпютърът** е свърхбързодействащ универсален компютър, извършващ паралелна обработка. Бързодействието му се измерва в брой операции за 1 секунда при обработка на 64-битови числа с плаваща запетая. Тази единица има означение MFLOPS . Някои суперкомпютри съдържат десетки хиляди процесори, симулиращи милиони клетки от човешкия мозък. Вероятно ще дойде времето, когато компютъра ще моделира мисловния процес на човека. Бурното развитие на микрокомпютрите промени смисъла на понятието "суперкомпютър". Съвременна работна станция, базирана на микропроцесор и разположена на бюро, има производителност, съизмерима с тази на суперкомпютър от осемдесетте години на миналия век.

Съвременните паралелни компютри се разделят на два големи класа в зависимост от това, доколко наследяват фон Ноймановата архитектура:

- а) паралелни архитектури, непосредствено еволюирали от фон Ноймановата архитектура – мултипроцесори, мултикомпютри, асоциативни процесори, систематични процесори, векторни процесори, паралелни компютри с функционална реконфигурация;
- б) паралелни компютри с принципно нови механизми за управление – паралелни компютри с управление от потоци данни, архитектури, ориентирани към езици за функционално проектиране и паралелни логически компютри.

1.8 Характеристики на компютрите

При избор на компютър се изхожда от неговото бъдещо приложение и от финансовите ресурси на купувача. За **пълното описание** на даден компютър е необходимо да се знае:

- **Наименование**, модел, евентуално годината на производство.
- **Предназначение**- универсален или специализиран (ако е универсален - клас-суперкомпютър, голяма машина, мини- или микрокомпютър)
- **Производителност** (в MIPS, MFLOPS, или оп./сек.)
- **Състав и характеристики** на хардуера- процесор(и), външна и вътрешна памет, канали, периферни устройства:
 - тип и производител на процесора;
 - работна честота на МП;
 - ОП и Кеш-памет;
 - BIOS;
 - видеоконтролер (видеокарта), звукова карта
 - мощност на захранващия блок;
 - конструктивно оформление на корпуса;
 - наличие на ПУ и техните основни параметри;
 - възможности за включване към мрежа;
 - вид на ситемната магистрала, която поддържа ПУ;
 - тип на клавиатурата;
 - видеомонитор, други ПУ.
- **Общи технически характеристики**- габарити, тегло, консумирана енергия, захранване.

- **Софтуер**- Операционна система, състав на системния и потребителски софтуер.
- **Надеждностни характеристики**- средно време за безотказна работа.
- **Икономически характеристики**- цена, стойност на консумативите при експлоатацията, гаранционните срокове.

Персоналните компютри са най-популярните и най-динамично развиващите се универсални микрокомпютри. Те от своя страна могат условно да бъдат разделени на две групи – преносими (наричат ги "**laptop**", "**notebook**") и "**desktop**" – стационарни компютри, предназначени за използване в помещения на офиси, лаборатории и т.н.

Полезно е да се знае какво означават съкращенията в разните оферти на доставчиците. В следващите примери с удебелен шрифт са написани офертите, а с нормален- обяснение за всеки параметър.

1. Примерна оферта за стационарен компютър (2006 г.)

TURBO X – това е наименование, модел

Intel CELERON D346 64 bit 3.06 GHz- Фирма – производител “ Интел” и тип на процесора Селерон D346,с тактова честота 3.06 GHz, 64 битова шина за данни
DDR2 512MB – оперативна памет (RAM) тип DDR2 с капацитет 512 MB
HDD 80GB 7200 rpm-“хард”диск с капацитет 80 GB и скорост на въртене 7200 об./мин.
DVD RW+/- Double layer- четящо/записващо DVD оптично устройство в два слоя
Case Middle ATX 400 W- корпус тип “средна кула”, захранващ блок 400 W
FDD 3.5” 1.44 MB- 3,5 инч. флопидисково устройство за дискети с капацитет 1,44 MB
ASUS s775 LAN Sound- тип на дънната платка с вградени мрежова и звукова карти
Card 128 MB on board- вградена графична карта с памет 128 MB

Вместо горния процесор, може да има друг:

INTEL CORE 2 DUO E6300 (1.86 GHz/ 2M)- Двухъдрен процесор на “Интел” с честота 1,86 GHz и обем на вътрешната кеш-памет ниво L1 – 2 MB.

Към офертата се добавят типа на клавиатурата, мишката, монитора и др. Дават се гаранционните срокове в месеци за цялата конфигурация и за нейни елементи

2. Примерна оферта за мобилен компютър (от края на 2006 г.)

Aspire AS5611AWLMI

1. CPU: Intel Core Duo processor T2050, 1.66 GHz, 2 MB L2, FSB533

2. Chipset: Intel 945PM Express

3. RAM: 512MB DDR2

4. Screen: 15.4 “ WXGA Acer CrystalBrite, 1280x800

5. Videocard: Intel Graphics, TV/VGA Out

6. Storage: 100 GB 5400 RPM, DVD+R/RW DL

7. Audio: speakers, mic in, line out

8. Wi-Fi 802.11 a/b/g, Bluetooth, LAN, Modem, 4xUSB 2.0,

5-in-1 card reader (SD/MMC/MS/MS PRO/xD)

9. Microsoft Windows XP Media Center Edition

Това се интерпретира така:

- 1- ЦП: производител “Интел”, двухъдрен процесор с тактова честота (бързодействие) 1.66 GHz, с вътрешен кеш от ниво L2 с капацитет 2 MB, поддържащ работна честота на основната шина за данни 533 MHz;
- 2- Тип на схемния набор (чипсета);
- 3- ОП тип DDR2 с капацитет 512 MB;
- 4- Плосък екран с диагонал 15.4 инча, стандарт WXGA, резолюция 1280x800 точки.
- 5- Графична карта на Интел с изход за телевизор или VGA монитор

- 6- Външна памет: хард диск с капацитет 100 , скорост на въртене 5400 об./мин, устройство за четене и запис върху двуслойни DVD+R/RW оптични дискове
- 7- Аудиосистеми: говорители, вграден микрофон, аудио изход
- 8- Възможност за безжична връзка съгласно стандарти 802.11 a/b/g, Bluetooth; вградена карта за връзка по кабел с локални мрежи, модем и 4 порта за USB 2.0, комбиниран четец за 5 типа картни паметни (от фотоапарати и др.)
- 9- Операционна система, която е инсталирана.

В офертите са възможни са и други означения:

- 6-cell Li-Ion Battery** – Литиево-йонна акумулаторна батерия с 6 клетки
- Integrated Fingerprint Reader-** Вграден четец на пръстови отпечатьци
- ATI Radeon Xpress 1150** – тип на графичната карта
- Storage 60GB IDE-** Твърд диск с капацитет 60GB и интерфейс IDE (стар стандарт)
- Storage 80 GB SATA-** Твърд диск с капацитет 80 GB и интерфейс SATA
- 2 x SO-DIMM socket expandable to 2 GB-** слотове за разширение на ОП
- 1.3 MP Orbitcam-** Web-камера модел Orbitcam с резолюция 1,3 Мегапиксела
- IEEE 1394a-** Високо скоростен интерфейс Fire Wire за кабелно свързване на видеокамера към компютъра
- Modem 56K-** телефонен модем със скорост на трансфер 56 Kb/sec.

Обикновено по вестници и списания се срещат и следния тип оферти за компютърни конфигурации, разбирането на които е по-трудно, например:

Система Home Max, Celeron D356J 3.33 Ghz, Board 775XFIRE-VSTA, 512 MB DDR2/533,1.44 FDD, 160 GB HDD WD SATA, KB Unitek 2207 silver, 7300LE 128 MB PCIE, DVDRW Sony 170A, A4T SWOP48, SB, LAN 10/100, Genius Q06S, Delux M302 350 W

Това се чете така: Компютърна конфигурация с име Home Max; микропроцесор на “Интел” Селерон модел D356J, с работна честота 3,33 Ghz; дъно тип 775XFIRE-VSTA (775 е типа на съединителя за ЦП), ОП с капацитет 512 MB тип DDR2 с работна честота 533 MHz; флоридисково устройство за дискети с капацитет 1,44 MB; хард диск с капацитет 160 GB тип Western Digital с интерфейс SATA; клавиатура тип Unitek 2207 (сива) ; видеокарта тип 7300LE с памет 128 MB и свързване към шина PCIE; записващо и четящо устройство за DVD-дискове тип Sony 170A; вградена звукова карта; вградена мрежова карта за скорости 10/100 Mbs; тип на мишката- Genius Q06S и тип на корпуса Delux M302 със захранване 350 W.

ТЕСТ

1. Изключете излишното:

- а) АЛУ
- б) УУ
- в) ОП
- г) ИП

2. За да се намали времето за изпълнение на програмите, като се обработват едновременно няколко инструкции, решението е:

- а/ повишаване тактовата честота на микропроцесора
- б/ увеличаване броя и големината на вътрешните регистри
- г/ използване метода на конвейера
- д/ предсказване на преходите

3. Класическите архитектурни принципи на компютъра са формулирани от:

а) Джон фон Нойман-Голдщайн;

б) Джон Атанасов-Бери;

в) Джон Моучли-Екерт;

г) Джон Екерт-Голдберг

4. При изключване на захранването на компютъра се губят данните, съдържаща се в:

а) оперативната памет;

б) твърдия диск;

в) BIOS;

г) флаш-паметта.

5. За следната компютърна конфигурация: Intel Celeron 700(128K Cashe), 64 MB SDRAM, 10GB HDD, 3,5" FDD, Monitor 15" EV 500, 52X CD-ROM, Fax modem 56K, Win 105 keyboard, MS Mouse, Speakers Creative GSC300, големината на оперативната памет е:

а) 64 MB;

б) 540 MB;

в) 8 MB;

г) 10 GB.

6. Машинен език е език, който:

а) се разпознава от процесора на компютърната система;

б) използва символни адреси и мнемоничен код;

в) обобщава много машинни инструкции в един оператор;

г) има един оператор за изчисляване на израз.

7. Кое не е означение на ОП?

а) EDO

б) DDR

в) DIP

г) FPM

8. Изключете излишното:

а) кеш- контролер

б) северен мост

в) западен мост

г) контролер на прекъсвания

9. Изключете излишното:

а) DDR

б) MBR

в) FAT

г) DIRECTORY

10. Тактовата честота на микропроцесора се измерва в:

а) GHz;

- б) kW;
- в) MB;
- г) Mb/s;

11. Стек се формира в:

- а) ОП;
- б) Хард диск;
- в) ROM;
- г) DVD;

12. Параметърът “ Резолюция “ не се отнася до:

- а) скенера;
- б) принтера;
- в) видеомонитора;
- г) клавиатурата;

Глава 2. ПРОГРАМНО ОСИГУРЯВАНЕ (СОФТУЕР)

2.1 Функции и класификация на операционните системи

Компютрите могат да работят и да бъдат използвани само ако са снабдени със съответното програмно осигуряване.

Програмното осигуряване представлява съвкупност от програмни продукти, програмни системи, документация и ръководства, осигуряващи ефективно използване на даден тип компютри от широк кръг потребители. **Програмен продукт** се нарича професионално изготвена **компютърна програма**, снабдена с документация, която е предназначена да се ползва от широк кръг потребители независимо от нейния създател и може да бъде предмет на покупко-продажба.

У нас широко се използва английското название **софтуер (software)**. С него се означава конкретното програмно осигуряване за даден компютър. Заедно с програмното осигуряване (ПО) се използва и понятието програмно-апаратно осигуряване- **фърмуер (firmware)**. Това са програми, които не се зареждат в компютъра от външен носител, а се намират в постоянната (ROM) памет.

Програмното осигуряване се разделя на две групи:

- **системно програмно осигуряване**, включващо операционни системи (ОС), езици за програмиране, инструментални среди и средства за програмиране, средства за техническо обслужване.
- **приложно програмно осигуряване** (специализирани програмни продукти, пакети приложни програми, бази данни и др.) Пакетите приложни програми представляват специализирани програмни системи със собствено управление и език за общуване с потребителите и всички програми, създадени за решаване задачите на потребителите.

Понякога се използва термина **базово програмно осигуряване** като еквивалентно название на системното програмно осигуряване.

Компютърната програма служи за управление на даден компютър при решаване на определена задача, той изпълнява действията, предписани му от тази програма. Най-често тя представлява описан на даден програмен език алгоритъм.

Програмата може да се състои само от няколко реда (написани за няколко минути от програмиста върху екрана, или на лист хартия) и чрез нея на екрана да се изпише "Здравей, това съм аз", или обратното - от хиляди или даже милиони програмни линии. Програмата за предполетен контрол на американска космическа совалка се състои от над 26 милиона програмни реда, написани за 22000 "човекогодини". Големите програми за индустрията се разработват от екипи от програмисти.

В началото на 70-те години се обособиха организации и фирми, разработващи програмно осигуряване, с използване на принципите на промишленото производство.. По-известни софтуерни компании в света са Microsoft, Novell, Borland, Lotus и др. Със своята продукция те участват пряко на софтуерния пазар. Възникнаха термините "софтуерно инженерство" и "автоматизирана система за създаване на софтуер" при разработване на много сложни програмни продукти.

Изключително важна роля за работата на компютърната система има най-важния компонент на системното програмно осигуряване – **операционната система(ОС)**.

ОС е съвкупност от програми, които управляват и осигуряват ефективното използване на апаратните и информационните ресурси на компютърните системи, при изпълнението на програми и улесняват използването им от потребителите. ОС е сред най-сложните софтуерни продукти. Тя е посредник между човека-потребител и компютъра.

ОС има следните **основни функции**:

1. Осигуряване на среда за изпълнение на потребителските програми.

При управление на програмите ОС:

- Поддържа работата на всички програми и осигурява взаимодействието им с хардуера.

- Осигурява взаимодействие на програмите с външните устройства и една с друга

- Обработка файлове

- Управлява трафика на данни

- Управлява потока от данни и процеси на пускане и спиране .

- Управлява на достъпа до диска и другите запомнящи устройства

Поддържа файлова система за съхранение и обработка на файлове.

2. Конкретното управление на апаратните и информационните ресурси на компютърната система:

- Управление на външната памет

- Управление на оперативната памет

- Осигуряване мултипрограмна работа

- Управление мултипроцесорна обработка на данните

- Работа в мрежа

-Откриване на различни събития в процеса на работа и осигуряване на съответна реакция

-Поддръжка на драйверен модел

- Поддръжка на **Plug&Play** спецификацията.

- Управление на захранването.

3. Интерфейс (връзка) между компютърната система и потребителя.

ОС предоставя на потребителя възможност за общо управление на компютъра с помощта на потребителски интерфейс, чрез който могат да се реализират различни

операции- създаване, копиране на файлове, изпълнение на програми, комуникации и др.

В зависимост от вида и предназначението си, конкретната ОС би могла да има и **други функции**.

Драйверите са специализирани програми за управлението на компютърните устройства (памет, екран, дискови устройства и др.). Инсталируеми са тези драйвери, които се включват при нужда при началното конфигуриране на системата. Те не са част от ядрото (драйвери с разширени функции за допълнителни физически и логически устройства).

Системните обслужващи програми (наречени “**утилити**”) се съхраняват като самостоятелни файлове върху диска.

Технологията Plug & Play осъществява автоматично конфигуриране на компютъра при добавяне на нов хардуер. Всяко поставяне на нова карта или устройство задвижва след включването на захранването процеса на преконфигуриране на ОС и ресурсите се преразпределят без намесата на оператора.

Всяка ОС има **системни изисквания** за типа на процесора и тактовата му честота, минимален обем ОП.

Мястото на ОС в компютъра е показано на фиг. 2.1.

| | | |
|----------------------|-----------------|--|
| Приложни програми | | Издателски системи, игри, бази от данни и други |
| Помощни Програми | | |
| ОС | Съставни модули | Редактори и др. |
| | | Асемблер, свързващи програми, компилатори |
| | | Команден интерпретатор (Обвивка) |
| | | Файлова система |
| | | Управление на В/ И |
| | | Управление на паметта |
| | | Ядро |
| Двоични програми | | Машинен език (0, 1) |
| Хардуер на компютъра | | Микропрограми |
| | | Микропрограмен код |
| | | Физически устройства |

Фиг. 2.1

Макар че операционната система е съвкупност от програмни модули, реализиращи отделните ѝ функции, не е необходимо цялата ОС да е заредена в оперативната памет. Това се налага само за някои от най-важните и най-често

използвани нейни компоненти, реализиращи функции, изискващи незабавно изпълнение. Съвкупността от тези елементи се нарича **ядро**.

Ядрото най-често реализира следните функции:

- управление на процесора (планиране на натоварването му и превключването му между изпълняваните програми);
- управление на прекъсванията;
- управление на изпълняващите се програми;
- управление на междупрограмните взаимодействия.

В началото на 50-те години от миналия век компютрите са били използвани от тесен кръг специалисти. Първите компютри не са имали ОС и никакви улеснения за потребителите, които са програмирали на машинен език. Програмите са били въвеждани ръчно - бит след бит от централния пулт за управление на машината, от там ръчно е ставало стартирането на програмите, там са били изобразявани и резултатите. Компютрите са обработвали последователно данните, липсвал е паралелизъм между входно-изходния обмен и обработката на данните от процесора. Един потребител е разполагал с цялата система за определен период от време и е имал пряк достъп до нея. Предимството тук е възможността за диалогова работа.

С развитието на компютърната техника са се променяли и програмните средства за улесняване работата на потребителите, и режимите за тяхното използване.

С внедряването на по-съвършени средства за автоматизация на програмирането (сложни асемблерни езици и езици от високо ниво) се наложило да се автоматизират процедурите за вход/изход (В/И) чрез специални програми. Създават се ОС за еднопрограмна (последователна) пакетна обработка.

В компютрите от трето поколение започва използването на нов метод за обработка на програмите - **мултипрограмирането** и нов метод за предоставяне на изчислителните ресурси на потребителите – **времеделенето** в многопотребителските системи. **Многопотребителска** е компютърна система, поддържаща едновременна работа на повече от един потребител.

Мултипрограмирането (многозадачността) е едновременното изпълняване на две или повече програми на един компютър. Времеделенето е режим на работа на многопотребителска изчислителна система, при който на всеки потребител последователно се предоставя неголям интервал от време. С това се постига усещането за това, че всеки един от тях работи самостоятелно със системата.

Друг тип ОС работят в **реално време** - действията им се управляват от външни събития, а реакцията на тях - бърза и навременна в рамките на необходимия интервал от време. Използват се в системи за управление.

Съвременните ОС са внедрени в **разпределените информационни системи** и компютърните мрежи. Развитието на ОС продължава и сега.

Класификацията на ОС може да бъде направена по следните критерии.

Според **предназначението**:

1. За индивидуална работа - осигуряват еднотребителска еднозадачна (възможно и многозадачна) работа - MS DOS, Windows.
2. За работа в реално време.
3. За обработка на транзакции - в системи за дистанционна обработка на много заявки към общ централен масив данни.
4. С общо предназначение - UNIX.

Според броя на обслужваните потребители и едновременно изпълняваните програми:

1. Еднотребителски еднозадачни - MS DOS.
2. Еднотребителски многозадачни - Windows

3. Многопотребителски многозадачни – UNIX, Linux, Windows XP.

Според начина, по който контактуват потребителя:

1. С команден интерпретатор - MS DOS, UNIX, Linux

2. С графичен интерфейс - Windows.

Огромно разпространение получиха версиите на операционната система Microsoft Windows. ОС Windows използват графичен потребителски интерфейс- **GUI** (прозоръчен интерфейс) за персоналните компютри, съвместими с PC. Графичния потребителски интерфейс) е способ за комуникация между потребителя и системата, който представя на екрана файловете чрез изображения, наречени икони, резултати и изображения, групирани в "прозорци", осигурява действията с тях чрез показалец, управляван от мишка или друг интерактивен манипулатор.

Между различните активни програми могат да се обменят данни (текст, числа и др.) чрез областта **Clipboard**. В тази област временно се съхраняват данните на различните програми .

Съвременните ОС работят с **виртуална памет** . Виртуалната памет е отделяне на потребителската логическа памет от реалната оперативна памет. Тя се реализира чрез различни методи. В ОС с виртуална памет програмистите съставят програмите си така, като че ли имат на разположение неограничена по обем оперативна памет.

Windows използва **технологията Plug&Play** (съкратено **P&P** или **PnP**), която включва:

- P&P хардуерни устройства;

- P&P ОС;

- P&P BIOS.

Базовата входно-изходна система – BIOS, управлява обмена на информация между хардуера и ОС. Важна особеност на P&P BIOS-а е, че той проследява работата на всички инсталирани устройства, което позволява на ОС да получава информация за всеки инсталиран компонент. Той не допуска конфликтни ситуации в хардуера.

Ако BIOS не поддържа P&P, той може да бъде обновен. Ако BIOS не е P&P-съвместим, настройката на новите хардуерни компоненти става с ОС.

В ранните версии на Windows бе въведен метода на корпоративната многозадачност- възможност за изпълнение на повече от една програма едновременно. По- късно бе въведена техниката на приоритетната многозадачност, премахваща недостатъка на корпоративната многозадачност- лошо написаните приложения да не "дават" компютъра на други програми. Windows насила взема управлението от дадено приложение и го предава на друго, редувайки това много пъти в секунда.

Новите програми на Windows използват **многонишковостта**, която е вариант на приоритетната многозадачност. Многонишковост означава едновременна обработка на няколко задачи, наречени **нишки**, в една и съща програма. Няколко задачи могат да бъдат обработвани паралелно и не е необходимо една от задачите да чака, докато друга завърши изпълнението си, за да се стартира.

Операционната система **Linux** се превърна в основна операционна система за по-голямата част от сървърите, върху които се гради Internet. Linux печели позиции и като интегриран софтуер за малките преносими устройства.

Всички средства на ОС, чрез които става възможна работата на ниво файлове, образуват нейната **файлова система**. Тя е основна подсистема на ОС и съдържа програмни модули, таблици и друга системна информация за работа с файлове, за организация и управление на ВЗУ, за управление на достъпа до файловете. Информацията, която е в даден файл, може да бъде програма или данни, които се използват от програми.

Според съдържанието си, файловете се разделят на програмни (изпълними, активни) – съдържат команди или инструкции, които могат да бъдат изпълнени от компютъра и текстови (пасивни) – съдържат текстова информация.

Според предназначението си файловете в MS DOS са обикновени (системни и потребителски файлове, драйвери и т.н), служебни – (директории ,каталози) и специални (част от ОС, предназначена за управление на устройство).

Чрез своята файлова система ОС дава възможност за работа с ВЗУ на логическо ниво, независимо от реалното физическо разположение на данните върху тях. За целта дисковото пространство се разделя фиктивно на логически структури, които облекчават достъпа до желаната информация. При по-голямата част от ОС се поддържа следната йерархия на логическа организация:

- логически блокове;
- клъстери;
- файлове;
- каталози (папки);
- дялове;
- устройства.

В ОС MS DOS и WINDOWS е приета система, при което всяко дисково устройство, мрежата и други устройства се означават с поредни букви от латинската азбука, например **C:** - за първи твърд диск, **A:** - за първо флопидисково устройство, **G:** - за мрежа и т.н. В рамките на всяко логическо устройство файловата система е йерархична тип “обърнато дърво”. Файловата структура на UNIX е подобна на MS DOS йерархична структура, но е по-еднообразна . Файловете, директорииите и даже входно-изходните устройства се третираат еднакво, няма концепция за "устройство" като в DOS, всички UNIX файлове "висят" от един корен.

Всеки файл е записан в ограничен брой физически блокове, които се наричат сектори (обикновено с размер 512 В), които не е задължително да бъдат разположени последователно в една писта. Поради това ОС трябва да има и да поддържа информация за физическото разположение на всеки файл върху информационния носител. ОС организират достъп до файловете и работят с ВЗУ на ниво няколко последователни блока, наречени клъстери. **Клъстерът** е комбинация от два или повече сектора, разположени един до друг върху една и съща писта. Това е най-малката част, която ОС използва за съхраняването на информация. Даже ако един файл се състои само от един байт, за неговото съхранение се използва цял клъстер.

Директорията е служебен файл, в който се съдържа информация за всеки файл в групата (име, дължина, дата и време на създаване и атрибут).

ОС MS DOS и WINDOWS имат средство бързо да откриват файл върху диска-таблица за разположение на файловете (FAT), която се намира на най-външната – нулевата писта от диска. Тя описва последователността от клъстери, принадлежащи на всеки файл и дава връзката между физическото разположение на информацията и логическите заявки от ниво на ОС. Ако съдържанието на FAT се разруши, достъпа до информацията на диска става невъзможен.

Друг специален файл, който се създава върху диска се нарича Главен каталог (Root Directory), и в него се пазят както имената на файловете върху диска, така и служебна информация за ОС (тип на файла, време и дата на създаване, размер и номера на първия клъстер на файла).

С цел зареждане на ОС върху дисковете се поддържат още 2 логически структури– главен запис за начално зареждане MBR, който липсва при флопи дисковете (MASTER BOOT RECORD) – цилиндър 0, глава 0, физически сектор 1 и запис за начално зареждане BR (BOOT RECORD).

Всеки от физическите твърди дискове може да бъде разделен логически на дялове (partitions). Това позволява върху едно устройство да се инсталират повече операционни системи, например WINDOWS и LINUX. В MBR се пази информация за всеки един от тях. Поради данните в него често MBR се нарича таблица за дяловете (partition table). Активен е този дял, в който има записани файлове, изграждащи ОС.

Записът и четенето на файл е сложен процес, в който участват приложения софтуер, ОС, BIOS и изпълнителния механизъм на дисковото устройство.

2.2 Операционна система UNIX

През 60-те години на миналия век заради увеличената изчислителна мощ на компютрите, възниква необходимостта от създаване на многопотребителска ОС с интензивен режим, времеделение на ресурсите между различните задачи, лесна за използване и с възможност за пренасяне на различни машини.

Развита през седемдесетте години на миналия век, UNIX стана една от най-използваните в промишлеността и компютърните мрежи операционни системи. UNIX се развива непрекъснато. Понастоящем съществуват около 100 версии, сред които IBM AIX, BDS/OS 2.0, HP-UX, SCO-UNIX, LINUX, X-Window System (или X) и др. Основните концепции на UNIX, доказали нейната жизнениост и удобство - дървовидната йерархична файлова организация, опростеното понятие за файл, именуването и третирането на периферните устройства като файлове, йерархичната система на активните програми (процеси), работата в мрежа и др., са внедрени в почти всички съвременни ОС.

Основните характеристики на UNIX са следните:

1. UNIX е многопотребителска и многозадачна ОС, работеща в режим на времеделение и мултипрограмиране. Освен това тя притежава елементи на ОС за работа в реално време.

2. При своята работа UNIX осигурява виртуално управление на паметта и устройствата, и автоматична буферизация на входа/изхода;

3. Информацията се съхранява в дървовидна файлова организация с пълна защита. Дървото на директориите е общо за цялата система и може при нужда да се разполага реално върху постоянни и сменяеми дискове.

4. Регистрирането на потребителите и поддръжка на системата (включване, изключване, профилактика, обновяване и др.) се осъществява от специалист, наречен системен администратор (root).

Всеки потребител на системата UNIX притежава своя домашна директория (НОМЕ-директория) с име, съвпадащо с неговия идентификатор. Тази директория се назначава от администратора и представява началната директория, в която попада потребителят при влизането си в системата. В своя НОМЕ-директория всеки потребител може да образува толкова поддиректории, колкото са му нужни.

Системата UNIX има следните предимства:

- написана е на език от високо ниво;
- лесна е за разбиране и модификация;
- мобилна (лесно пренасяне върху други компютри);
- съдържа средства за създаване на големи програмни системи;
- има проста и надеждна файлова система;
- удобен интерфейс за връзка с периферните устройства;
- многопотребителска и многозадачна ОС система;

ОС UNIX се състои от малко ядро с необходимите утилити за обслужване на файловата система и за изпълнение на програмите в многозадачен режим, и от т.н. обвивка

| |
|-------------------------|
| Потребител |
| Обвивка (Shell) |
| Ядро |
| Хардуер |

Ядрото на UNIX е написано почти изцяло на езика C и това прави тази ОС машинно независима и мобилна, т.е. приложима практически за всички видове компютри - от преносими до суперкомпютри, еднопроцесорни и многопроцесорни.

Ядрото има 2 компонента със следните функции:

а) система за управление на процесите - осигурява режима на времеделение, т.е. възможността за едновременно изпълнение на няколко приложни програми с еднопроцесорен компютър, управление на процесите и комуникациите между тях (осигурява мултипрограмната и многозадачната работа);

б) система за управление на файловата система - съвкупност от специално организиран набор от данни върху външните устройства на компютъра и програмни средства, гарантиращи достъпа до тези данни и тяхната защита.

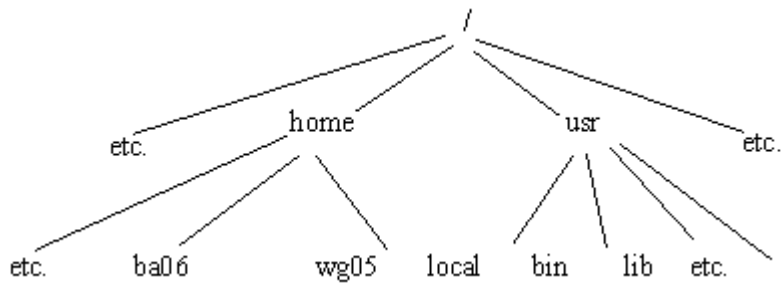
Ядрото комуникира директно с хардуера, а между ядрото и потребителя се намира обвивката (в Unix тя се нарича Shell. Тя осигурява връзка на потребителя с набор от команди, които могат да се изпълняват след въвеждане от клавиатурата или прочетени от файлове (подобно на файловете BAT от MS DOS) . Такива файлове, съдържащи команди на обвивката, се наричат шел-скриптове (Shell scripts)или Shell-файлове. Те са написани от потребителя и му позволяват да дефинира свои по-сложни команди към ОС. Обвивката на UNIX е командния интерпретатор на системата - редовете от клавиатурата и изпълнява разположени в зависимост от това какво е въведено.

Всеки UNIX има свое съответно копие на shell и може да работи, без да пречи на другите клиенти.

UNIX има свой графичен интерфейс, наречен X-Windows по подобие на Windows.

Понятието "файл" в UNIX се определя като линейна от байтове (символи), завършваща със символа EOF . В UNIX има три вида файлове: обикновени (системни и потребителски), файлове-директории и специални файлове (драйвери на периферни устройства). Особеност на UNIX е, че тя разглежда всички периферни устройства (терминали, принтери, дискове) като файлове и процедурата за достъп до тях е аналогична на процедурата за достъп до файл.

Файловата структура на UNIX е подобна на MS DOS йерархична структура, но е по-еднообразна . Файловете, директорииите и даже входно- изходните устройства се третират еднакво (с изключение на невъзможността за пряк достъп за четене от принтера). В Unix няма концепция за "устройство" като в DOS , всички UNIX файлове "висят" от един корен, означен с / - root. Концепцията за път също съществува в UNIX, но за разлика от DOS сепараторът за път е /, а не \ . Примери за означаване на пътища: /home/ba06/msc,/usr/local/work



DOS/NT файлова структура

Основни команди на ОС UNIX

Системата UNIX е чувствителна към въвеждането на командите с малки или големи букви. Потребителят трябва да въведе своя идентификатор (id) и паролата си с малки букви (долен регистър на клавиатурата), в противен случай UNIX ще счита, че оттук нататък потребителя ще въвежда всички команди с главни букви (от горния регистър на клавиатурата).

При работа системата издава следните подканящи съобщения:

login - очаква логическо включване на потребителя в системата. Трябва да се отговори с въвеждане на потребителския идентификатор (id) и след това да се натисне клавиша ENTER (този клавиш трябва да се натиска и след всяко ново въвеждане на команда от клавиатурата).

password - очаква се въвеждане на паролата, съответстваща на дадения потребителски идентификатор.

След влизане в UNIX, като резултат на екрана излиза промпта и чака за въвеждане на команда. Когато потребителят е включен в системата, той е в директорията home. Това съответства на устройството G: в компютърната мрежа NT .

При работа със системата се използват следните общи команди:

logout - логическо изключване от системата (не трябва да се забравя).

map - извежда информация за командата, чието име е изписано след тази команда.

help - команда за помощ, извежда списъка и синтаксиса на всички команди.

date - извежда датата и астрономическото време.

who - извежда на екрана списъка на всички потребители, които в момента са в системата.

top - тества работещите програми.

clear - изчиства екрана.

Много от командите в UNIX, например за преместване, създаване или премахване на директории, са подобни по формат и синтаксис на тези в MS DOS.

UNIX командите имат вида:

`command [-options] [parameters] [redirection]`

Квадратните скоби [] означават, че дадената част е опционна, т.е. може да присъства или не. Опционните части модифицират действието на командите. При написването си те трябва да се предхождат от минус (-) .

Следват някои полезни команди от системата UNIX.

`cal [month] year`

Отпечатва календар за зададената година (year) и за месеца (month - ако е специфициран). Месеца се задава с числата от 1 до 12.

`cat file ...`

`more file ...`

`less file ...`

Всички тези команди прочитат файл с име file и го изпращат към стандартния изход (това е екрана). Между имената на файловете трябва да има интервали. Може да се извърши пренасочване на изхода.

`cc [option] file ...`

Компилира файлове, с име file написани на езика C. Тези файлове трябва да са от типа .c. По подразбиране се създава изпълним файл a.out. Чрез опцията -o на крайния изходен файл може да бъде дадено друго име, например `cc -ofred.out fred.c`.

`cmp file1 file2`

Сравнява двата файла с имена file1 и file2. Тази команда дава съобщения в кои линии и в кои символи се различават двата файла. Ако не са изведени съобщения, двата файла са еднакви.

`cp file1 file2`

`cp file1 ... directory`

Копира файлове. Първата команда се копира file1 в file2, без източника да се изтрива. Втората команда копира списъка от файлове, започвайки от file1, в посочената директория.

`echo [-n] [arg]`

Извежда на екрана аргументите, следващи след командата до края на реда. Много полезна е за използване в командните shell-файлове (шел-скриптове). Ако е специфицирана -n, не се генерира нов ред след командата echo.

!!

Повтаря последната команда.

!a

Повтаря последната команда, започваща с a.

lpr [option] [name]

Предизвиква подреждането и изпращането на копия на посочените файлове за отпечатване. Опциите могат да определят кои принтери ще се ползват.

ls [option] [name]

Извежда на екрана съдържанието на директорията (подобна е на dir в MS DOS). Ако е зададена опция - l, се извежда не само името на файла, но и неговите параметри (тип, собственост, размер, последна дата на модифициране).

mkdir dirname ...

Създава нова (нови) директории със зададено име (имена).

mv file1 file2

mv file ... directory

Преместване на файлове. Работи като cp, но изтрива изходния файл (файлове).

pwd

Отпечатва името на текущата работна директория.

passwd

Смяна на паролата.

rm file ...

rmdir directory ...

Изтриват съответните файлове и директории. Изискват потвърждение преди да се изпълнят.

vi name ...

Стартира екранния редактор vi .

wc [-lwc] file

Преброява думите във файла. Брой линии, думи или символи в зависимост от зададените опции. Ако няма такива, извежда информация за всички.

Специално внимание в UNIX е отделено на комуникациите. В рамките на една и съща система са възможни интерактивен разговор между двама потребители посредством клавиатурата и екрана (команди write, talk), автоматична напомняща служба (команда calendar), новини, електронна поща (команди mail, mailx, vmail, sendmail).

Mail - команда за електронна поща

Синтаксис:

- 1) mail потребител 1 [потребител2 ...]
- 2) mail система 1 [/]!система 2[/]!...системаN[/]!потребител
- 3) mail потребител@ [[[подобластN.]...] подобласт 1.] област
- 4) mail [опция] [файл-поща]

Освен това са възможни комуникации и с отдалечени системи. За да комуникират две UNIX системи, те трябва да са свързани чрез съобщителна линия .

Някои от командите за Internet - комуникации са:

ftp IP адрес или DNS-име

Прехвърляне на файлове . След влизането в ftp - сървър се появява промпта ftp> , и могат да се въведат команди като put (прехвърляне на файл към избрания адрес), get (трансфер на файл от избрания адрес към системата на потребителя), bye (излизане от сървъра) и др.

telnet

Отдалечен достъп (вход) в компютър, зададен с IP адрес или DNS-име. Връзката се прекратява с exit или logout.

ping

Проверка дали компютър с указаното име след командата е достъпен чрез мрежата с потребителя.

Всички файлове в имат набор от разрешени операции, които могат да се изпълняват върху тях. Те са четене (read-r), запис (write-w) и изпълнение (execute-x). UNIX също определя кой може да има достъп до всеки файл. Потребителят може да дефинира достъпа за три възможни групи собственик (Owner), група на собственика (Group), всеки друг (Everyone) . Собствеността върху файла е важна, тъй като тя определя разрешените действия с него. Ако има проблеми, свързани с това, че потребителя не е собственик, трябва този файл да се копира чрез sору в друг, собствен, и последния да се използва.

Всяка от горните групи чрез кода за защита на файла може да получи правото да чете, записва или изпълнява тези файлове. Кодът за защита има следния вид rwxrwxrwx .

Тези 9 букви могат да се разделят на три групи по три букви, специфициращи разрешение за четене - r, запис - w и изпълнение - x , съответно за собственика, групата му и за всички останали.

Замяната на някоя от тези букви с тире означава, че за дадения файл съответното право не е дадено.

За да се види кода за защита на файл, трябва да се напише

ls -l

тогава след промта ще се изпише, например

```
-rwxr--r-- 1 ba06 staff 2586 Oct 20 Poem.txt
```

Първата позиция показва дали това е директория или не - в случая това е файл заради (-). Следват три групи по три букви, специфициращи разрешение за четене - r, запис - w и изпълнение - x , съответно за собственика, групата му и за всички останали.

owner = rwx

```
group = r--
everyone = r--
```

След кода за защита цифрата 1 показва броя на файловете връзки, ba06 е потребителския идентификатор, staff показва че потребителя е от щатния състав, 2586 е броя на байтовете, Oct20 е датата на последната модификация на файла, а Poem.txt е името му.

Промяна на достъпа до даден файл, например с името sh1.ksh, може да се направи с командата chmod, със използване на следните стойности за операциите:

```
Read = 4
Write = 2
Execute = 1
```

Ако собственика на файла иска само той да извършва четене, запис и изпълнение, той трябва да събере $4 + 2 + 1 = 7$ и да запише 7 на първо място след chmod, а за групата на собственика и за всички останали да запише 4, което им разрешава само да четат този файл. Тогава синтаксиса на командата ще изглежда така:

```
chmod 744 sh1.ksh
```

ОС UNIX изпълнява два вида файлове. Първите, изпълнимите, обикновено съдържат изпълним код, получен обикновено след транслацията на програмата от език от високо ниво (C или Pascal) в машинен (двоичен) код. Вторите, наречени shell-файлове или shell-скриптове, са написани на езика на обвивката Shell. Тези команди се четат и изпълняват от обвивката. Чрез комбиниране на утилитите на ОС с пренасочването и останалите shell- скрипт файлове, от потребителя могат да бъдат създадени нови по-сложни средства в системата.

Както в MS DOS, изпълнимите файлове в UNIX се стартират чрез написване на техните имена след командния промпт. Тези файлове могат да се обръщат за данни към други файлове (или да четат, или да записват данни). Обаче всяка изпълнявана програма под управлението на UNIX се отваря автоматично за два потока от данни - стандартен вход и стандартен изход. Програмата чете и записва в тези потоци, ако не е специфицирано да взема или изпраща от файл. Обвивката Shell нормално насочва стандартния вход към клавиатурата, а стандартния изход - към екрана, но потребителя може да пренасочва стандартния вход или изход към файл или конвейер. Символите <i> се използват за пренасочване съответно на стандартния вход и изход към файлове.

Например ако програмата hello.out се изпълнява нормално, тя изпраща изхода към екрана. Обаче следващата командна линия

```
hello.out >bill.txt
```

предизвиква пренасочване на изхода от програмата към файла bill.txt, а на екрана няма да се изобрази нищо.

Пренасочването на изхода може да бъде изпълнено чрез системни UNIX команди (които също са програми). Така командата ls нормално изпраща своя изход към екрана, но

```
. ls >dir.txt
```

го изпраща към файла dir.txt. Пренасочването на входа става по подобен начин, но изисква по-голямо внимание. Например програмата wordcount, когато е стартирана, изисква името на файла, който трябва да бъде анализиран. Ако името на файла се помести в друг файл, то входа ще бъде взет от този файл, а не от

клавиатурата. Например ако файла rdin.txt съдържа само един ред - poem.txt, то командата

```
wordcount.out <rdin.txt
```

ще изобрази броя на думите и изреченията от файла poem.txt.

За комбинирано изпълнение на повече команди на системата, изписани в един команден ред, те трябва да се разделят с вертикална линия една от друга. Ако е необходимо, например, да се сортира по азбучен ред съдържанието на директория, най-бързия начин е да се издаде командата ls и да се "вкара" изхода и директно в командата sort.

```
ls|sort >sorted.dir
```

Това ще изведе в sorted.dir сортиран списък на съдържанието на текущата директория.

Програмиране с езика на обвивката

Обвивката е и команден, и програмен език, който осигурява възможности за изграждане на подходяща среда за потребителя и неговата група. Тази среда може да свързва в едно приложения, които нормално би трябвало да се инициират чрез отделни команди от команден ред (подобни са BAT-файловете в DOS). Например тази среда може да свърже компилатора и редактора така, че ако възникне грешка при компилирането, управлението ще се върне автоматично към редактора с цел коригиране на изходния текст.

Като програмен език, обвивката осигурява по-големи удобства за работа, отколкото обикновената UNIX система. Файлът, който съдържа shell-команди се нарича shell-файл или shell-script.

Shell-файловете се използват за:

- съхраняване на често използвани последователности от команди или такива с неудобен синтаксис осигуряване на интерактивна среда за потребителите или проектантите на софтуер;
- написване на backup-програми;
- инсталационен запис и конфигуриращи скриптове за инсталиране на нови продукти (еквивалентни на setup на Windows);
- бързо прототипиране за постигане бързина в работата;
- инициализация на UNIX;

Най-разпространени са три обвивки с текстов интерфейс:

- обвивка на Бурн (sh или bash), с команден промт \$,
- C-обвивка (csh), с команден промт %,
- обвивка на Корн (ksh), с команден промт #.

Всички тези обвивки имат общо ядро. Те се различават незначително по функциите си и синтаксиса на командния език, и имат приблизително по 200 команди. C-shell е малко по-добра от обвивката на Бурн при интерактивна работа, но с нея малко по-трудно се програмира. Сравнително по лека за програмиране на shell-файлове е обвивка на Корн (ksh).

По-новите UNIX системи (AIX на IBM, X-Window System, UNIX на SCO и др.) предлагат богат графичен интерфейс за работа с потребителя и възможност за общуване на около 60 различни езици, включително и български.

За да се създаде shell-файл, трябва да се напишат съответните команди във файла, да се даде на този файл разрешение за изпълнение чрез командата chmod, и след това

този файл да се стартира чрез съответния език на обвивката, като за целта в командната линия се напише неговото име. В следващите примери се използва обвивка на Корн (ksh).

Пример1 :

Името на файла е sh1.ksh, макар, че по принцип може да бъде всякакво. Съкращението .ksh се използва с цел ясно да личи, че това е Корн shell-файл, но това не е файлово разширение, което запуска обвивката на Корн за изпълнение на shell-файла.

- sh1.ksh

```
#!/bin/ksh
echo "Please enter your name: "
read name
echo "Your name is" $name
```

След като се даде на този файл разрешение за изпълнение и се стартира, неговото действие е следното:

```
Please enter your name:
Peter
Your name is Peter
```

Файлът трябва да започне с реда `#! pathname` (името на пътя в случая е `/bin/ksh`), което показва на системата да стартира обвивката на Корн. Това е програмната линия, която определя коя обвивка да се използва за изпълнение на shell-файла. Ако този ред липсва по подразбиране се стартира обвивката на Бурн.

Символът `#`, използван на всеки друг ред в програмата, освен на първия, показва че след него следва коментар. Например

```
# this is a comment line
```

командата `echo` е оператор за отпечатване,

```
echo hello
```

всяка дума след оператора `echo` се отпечатва на нов ред. За отпечатване на празна линия или достигане на нов ред трябва да се напише оператора `echo` без следващ текст, например

```
echo
echo hello
echo
```

ще отпечата празни линии преди и след думата `hello`.

Операторът `read` се използва за прочитане на клавиатурата и съхранение на въведената стойност в паметта заделена за дадената променлива, например

read name

където read е команда изпълнявана от shell-файла, а name е променливата която съдържа входната информация. За обръщение към променливата в програмата, трябва да се използва знакът \$, например,

echo \$name

Пример 2 : users.ksh

```
#!/bin/ksh
echo there are $(who | wc -l) users on the system
```

Изпълнението на файла дава:

There are 21 users on the system .

Файлт се използва просто за да с изпълни UNIX-команда от една линия, която е неудобно да бъде отпечатана по обикновения начин.

who - Извежда списък на потребителите, намиращи се в момента в системата.

wc -l - Преброяваща програма, която брои броя на линиите и следователно броя на потребителите (тъй като who извежда един потребител на една линия).

\$ (expression) - нарежда на Korn shell-интерпретатора да изчисли израза в скобите (expression) .

При написването на други програми за shell script-файлове могат да се използват операторите for... in ... do... done..., while... do...done...,if... else..., case, които се интерпретират от обвивката.

В системата UNIX има текстови редактори vi и jove, но те не са с дружелюбен интерфейс.

При написването на shell-скриптовите програми е удобно да се ползва редактор от типа на Textedit.

ОС Linux

Операционната система Linux вече се превърна в основна операционна система за по-голямата част от сървърите, върху които се гради Internet. Тази алтернатива напредва с бързи крачки и в областта на корпоративните мрежи и се превръща в най-разпространената операционна система в тази област. Linux печели позиции и като интегриран софтуер за малките преносими устройства.

Linux се базира на операционната система Unix, по-специално на Minix (още наричана Baby Unix), той създава собствен кернел (ядро на операционната система) и го прави с отворен код (open source), тоест свободно достъпен за всички, които искат да пишат в програмата. Отвореният код има това предимство, че човек сам може да модифицира своята операционна система спрямо нуждите си.

Благодарение на отворения код съществува постоянна възможност за разработване и модифициране на ядрото от различни групи хора. По този начин Linux постигна основното си предимство - разработчиците на Linux са гъвкави и предлагат постоянно нови идеи в областта системните на решения.

Още от самото начало Linux е бил замислен като многопотребителска операционна система. Концепцията за защита на системните ресурси, както и

предпазването на потребителите един от друг лежат в основата на програмния дизайн на Linux. За разлика от Windows 95/ 98/ Me, където само един потребител може да използва системата и обикновено има пълен контрол върху нея, в Linux потребителите са разделени на потребителски групи, всяка от които може да има ограничен достъп до определени ресурси (дисково пространство, процесорно време, външни устройства и мрежови ресурси). Единственият потребител с неограничени права за достъп е системният администратор .

Linux е съвкупност от ядро (кернел), GUI (графичен потребителски интерфейс) и често употребявани програми в системата. Тя се нарича дистрибуция. Тази операционна система е функционираща, защото предлага широк кръг приложения, работещи с нея офиспакета, няколко браузъра, версия на ICQ за Linux, версии на игри за Linux и т.н.

Благодарение на своя произход от UNIX свободната операционна система е създадена, за да обслужва клиенти в мрежа. Самият начин на работа гарантира по-висока стабилност - Linux предлага много по-висока степен на защита на програмите една от друга, което означава, че при срив при една от командите не увисват всички останали, а е възможно изключването на "забилия" процес, като останалите продължават да функционират.

Благодарение на принципа на отворения код Linux е операционна система, която на практика може да бъде използвана навсякъде.

Като отворена, базираща се на общодостъпни стандарти, системата Linux поддържа разнообразни протоколи, свързани със сигурността и достъпа до системата на потребителите. Пример за такъв стандартен протокол е Kerberos. Този протокол позволява изграждането на компютърни мрежи с централизиран сървър за достъп. При осъществяването на връзка между два клиента в мрежата те трябва да се свържат с централния сървър и да получат одобрението му. Това позволява изграждането на мрежи с много висока степен на контрол от страна на системния администратор, което играе решаваща роля за нивото на сигурност в цялата система.

Linux не е единствената съществуваща алтернатива на Windows и не е единствената операционна система с отворен код, но тя е най-популярната и най-добре развитата по отношение на предлаган софтуер и хардуерна поддръжка.

2.3 Езици за програмиране

Езиците за програмиране са средство за кодиране на алгоритмите. Описаният чрез даден език за програмиране алгоритъм представлява програма, която се изпълнява на компютъра. **Компютърен програмен език** е множество от символи и правила за тяхното използване, предназначени за управление действието на компютъра. Всеки език притежава своя собствена граматика и синтаксис, свой собствен начин за използване на идеи.

Нека да е необходимо да бъде изчислен простия математически израз $x = (a+b) - (c+d)$. Фрагментът от компютърна програма за изчисляване на този израз, написан на **език за програмиране от високо ниво**, изглежда така:

```
.....  
    x = (a+b) - (c+d);  
.....
```

Това е разбираемо за потребителя, защото е почти същото като математическия израз. Този ред се нарича **оператор**. Операторът е основен градивен елемент в езика за

програмиране за постигане на определени цели и функции. Една програма по същество е поредица от оператори.

Съвкупността от оператори, които реализират машинно изпълнение на даден алгоритъм образуват програмите. Програмите, написани на даден език за програмиране трябва да се преведат на **машинен език (код)**, който е единствено разбираем за определен тип компютър. Езикът за програмиране се състои от две части - съвкупност от правила за запис, определящи синтаксиса на коректно написана програма, и програма, която преобразува програмата на език от високо ниво в поредица от машинни инструкции, разбираеми от конкретен тип компютър с помощта на **транслатор**. Това е програма, която превежда изходната програма в машинен код. Според начина, по който се осъществява това превеждане, транслаторите са компилатори и интерпретатори. **Компилаторът** чрез непрекъснат процес прекодира целия текст на една програма от високо ниво, като създава нова програма на машинен език. Същата може да се съхрани и след това многократно да се изпълнява, без да се използва компилатора.

Работата с език, реализиран чрез компилатор, обикновено минава през три етапа: текстът на програмата се създава с помощта на текстов редактор или програма за обработка на текст; вторият етап (след като програмата е тествана и са отстранени грешките) се състои в компилиране на програмата и третият етап е самото изпълнение на готовата програма. Главното предимство на компилирането е бързината.

Интерпретаторът изпълнява програмата оператор по оператор, като още в момента трансформира всеки оператор в машинна инструкция. Това помага на програмиста да прави промени и веднага да вижда резултата, т.е. улеснява се процеса на писане и тестване на програмите. Основен недостатък на интерпретаторите, че изпълняват програмите много бавно

Разликата между компилатора и интерпретатора е аналогична на разликата между преводача на писмен текст и преводача на устна реч. Преводачът на писмени произведения разполага със завършен ръкопис и от него прави текст на друг език. Устният преводач предава всяка фраза или изречение непосредствено след изговарянето.

За горния израз фрагментът от машинната програма, записан в оперативната памет на микрокомпютъра в двоичен код изглежда така:

```
00000011 11000011
00000011 11001010
00101011 11000001
```

За човека този фрагмент е трудно разбираем, но само в този вид той се възприема от микропроцесора на Intel и може да се изпълни от него.

На първите компютри от 50-те години на XX век е програмирано на машинен език. Той е единствен, който може да се използва директно от компютъра. Машинните езици са езици от първо ниво по отношение апаратурата на компютъра. От гледна точка на програмиста е трудно да се работи с тях, защото се използва двоична бройна система и следствие на това голяма е вероятността за допускане на грешки. Машинните езици за различни типове компютри са различни и следователно не е възможна преносимост на програма от един за друг тип компютър. Научаването на един машинен език не позволява естествен преход към усвояването на друг машинен език. Поради тези съображения специалистите насочват своите усилия към разработването на по-съвършени системи за програмиране.

Асемблерните езици се появяват в началото на петдесетте години с цел да се намалят трудностите при писане на програми на машинен език. В асемблерния език

мнемоничен код заменя кода на операцията, а вместо абсолютни адреси на клетки се използват символни адреси.

Горният фрагмент от машинната програма, преписан на асемблера на микропроцесор на фирмата Intel изглежда така:

```
add ax, bx
add cx, dx
sub ax, cx ,
```

където **add** е мнемонично съкращение от addition (събиране) , **sub** - от subtraction (изваждане), а **ax**, **bx**, **cx** и **dx** са имената на регистри на микропроцесора, в които има операнди и резултати. Първият ред означава да се събере съдържанието на регистри с имена **ax** и **bx** и резултатът да се помести в регистър **ax**. За да бъде “разбран” и изпълнен от компютъра, този текст трябва да се обработи или транслира. Това прави специална програма, наречена **Асемблер**, която транслира (превежда) изходната програма в специален вид, наречен обектен (машинен) код. При това преобразуване една инструкция в изходната програма съответства на една инструкция в машинния код. За спестяване на програмистки труд се използват макроасемблерите, в които се дефинират и наименоват групи от често изпълнявани инструкции, наречени макроси. При транслиране на машинен език един макрос се заменя с цяла група машинни инструкции. Асемблерният език също като машинния е различен за всеки тип микропроцесор. Сега тези езици се използват главно за създаване на системно програмно осигуряване.

При решаването на конкретна задача на компютър от някаква предметна област е естествено да се работи с понятията на тази област и да се състави програма, съответстваща на използваните термини. Развиването на тази идея води до създаване на **езици за програмиране от високо ниво**(понякога ги наричат алгоритмични езици), като FORTRAN, BASIC, COBOL PASCAL, C++, Java и др. С тяхната поява се цели да се елиминира отегчителната и склонна към грешки природа на асемблерния език. От гледна точка на програмиста тези езици постигат независимост от конкретния компютър. Програмистите не се интересуват от разпределението на компютърните ресурси и се грижат само за създаване на ефективен алгоритъм за решаване на конкретната задача.

В индустрията се използват езиците C, C++ и Java. За научни изследвания се използват Fortran, LISP, PROLOG и др.

Според възможността за достъп до изходния код на програмата, написан във вид на оператори като текстов файл, има софтуер със **затворен код, отворен и свободен код**.

Софтуерът е със затворен код, когато е наличен само изпълнимия му код. Повечето комерсиални програмни продукти се разпространяват по този начин..Отворения код означава възможност за достъп до изходния код при определени условия . Софтуерът със **свободен код** позволява на клиента да прави с изходния код и програмата каквото си иска. Такива са приложенията под ОС Linux.

. Дейността по създаването, проверката и документирането на програми се нарича **програмиране**. Хората, които професионално се занимават с програмиране, се наричат **програмисти**, а лицата, които използват програмите за да решават с тях свои задачи са потребители.

Процесът на разработване на програмата преминава през няколко етапа:

- съставяне на математически модел на задачата и логическа схема за решаване на проблема;

- логическа схема на компютърната програма;
- съставяне на компютърната програма;
- проверка и корекция на готовата програма (тестване на програмата);
- съставяне на документация на компютърната програма.

Първият етап е съставяне на математическия модел и логическа схема за решаване на проблема от програмата. В този етап програмистът уточнява формули, входни и изходни данни за програмата, уточнява по какъв начин ще бъдат обработвани данните от програмата. Главно внимание се отделя на реда и последователността на въвежданата в компютъра информация. Определя се последователността на извършване на основните операции в процеса на обработка. Логическата схема има по-общ характер и отделните операции не се разглеждат подробно.

Вторият етап е съставяне на логическа схема на програмата. През този етап се написва подробно алгоритъма, по който ще се извършва обработката на данните. Най-често алгоритъмът се описва във вид на блок-схема.

Третият етап е етап на съставяне на програмата. Отделните блокове на блок-схемата се превръщат в поредица указания чрез езиците, които се използват за описване на програми и се наричат **езици за програмиране**. В процеса на създаване на програмата трябва да се обърне специално внимание на изработването на интерфейса – това е онази програмна част, която реализира общуването между потребителя и програмата. Самият интерфейс трябва да бъде “дружелюбен” с потребителя, т.е. да го подпомага в работата с компютърната програма.

Четвъртият етап- проверка на готовата програма, е най-тежкият етап в процеса на създаване на програмата. Откриват се и се коригират грешките, които могат да се допуснат в програмата- синтактични и логически. Синтактичните грешки са следствие от неправилно използване правилата на съответния програмен език. Тези грешки се откриват много бързо при транслирането на програмата (превеждането ѝ в машинен код). Логическите грешки са трудно откриваеми, тъй като те са следствие от грешки в логиката на алгоритъма. През този етап програмата се изпълнява с примерни данни, за които са известни резултатите, които трябва да се получат.

Последният етап е съставяне на документация за разработената компютърна програма. Документацията трябва да съдържа: описание на предназначението на програмата, описание на алгоритъма, самата програма и ръководство на потребителя. Ръководството на потребителя пояснява как се работи и как се използва съответната програма. То трябва да бъде точно, ясно и да съдържа указания за работа при всички възможни реакции на разработената програма.

От началото на 70-те се развива **структурното програмиране**. Основен подход в него е използването на модули в писането на програми. **Модул** е съвкупност от оператори, които се разглеждат от едно цяло. Програмата е разбита на по-малки единици-модули, които са самостоятелни програмни единици със собствено име и определена функция (процедура).

При създаването, използването и поддържането на програмното осигуряване работят следните специалисти: програмисти, потребители и експлоатационен персонал. В зависимост от вида на създаването програмно осигуряване програмистите са системни или приложни (създаващи приложно програмно осигуряване). Потребителите на приложно програмно осигуряване са специалисти в различни области.

Създаването на дадено приложение най-общо преминава през няколко стъпки, като за всяка от тях са необходими съответни инструментални средства. Към тези средства като минимум могат да се включат: текстови редактори, графични редактори, транслатори, програми за настройка и тестване на приложенията и др.

Текстовите редактори се използват за създаване и изменение на текстови файлове. Текстът може да бъде последователност от оператори на някакъв език за програмиране, оформен като програмен модул. За целта се прилагат екранни и редови текстови редактори, като по-удобни са екранните.

Следващата стъпка в процеса на разработване на потребителски програми е обработването на създадените с текстов редактор файлове с помощта на транслятор. Това е програма, която превежда изходната програма в машинен код. Според начина, по който се осъществява това превеждане, трансляторите са компилатори и интерпретатори. **Компилаторът** чрез непрекъснат процес прекодира целия текст на една програма от високо ниво, като създава нова програма на машинен език. Същата може да се съхрани и след това многократно да се изпълнява, без да се използва компилатора. Работата с език, реализиран чрез компилатор, обикновено минава през три етапа: текстът на програмата се създава с помощта на текстов редактор или програма за обработка на текст; вторият етап (след като програмата е тествана и са отстранени грешките) се състои в компилиране на програмата и третият етап е самото изпълнение на готовата програма. Главното предимство на компилирането е бързината

Интерпретаторът изпълнява програмата оператор по оператор, като още в момента трансформира всеки оператор в машинна инструкция. Това помага на програмиста да прави промени и веднага да вижда резултата, т.е. улеснява се процеса на писане и тестване на програмите. Основен недостатък на интерпретаторите, че изпълняват програмите много бавно. Интерпретаторът трябва да се използва всеки път, когато се изпълнява програмата и по този начин се заема повече памет отколкото при изпълнение на компилираните програми.

Разликата между компилатора и интерпретатора е аналогична на разликата между преводача на писмен текст и преводача на устна реч. Преводачът на писмени произведения разполага със завършен ръкопис и от него прави текст на друг език. Устният преводач предава всяка фраза или изречение непосредствено след изговарянето.

. Следващата стъпка на разработване на потребителски програми е свързването. За изпълнение на тази стъпка се използват **свързващи редактори**, които обединяват в една обща изпълнима програма няколко обектни модула и библиотеки.

Настройката е важна стъпка от процеса на разработването на потребителски програми. Могат да се използват различни средства за настройка. **Дебъгерът** (Debugger, програма за настройка) е специална програма, която подпомага настройването и тестването на потребителските програми. Тя предоставя средства за **трасиране** (стъпково изпълнение на програмата и запис на последователността от действия по време на изпълнение). Тя може да се включва в или като част от компилатора или интерпретатора и помага на програмистите да откриват и коригират грешките.

Често отделните програми, включени в групата инструментални средства се обединяват в т.нар. **интегрирани среди**. Това понятие доби популярност при създаване на програмно осигуряване, работещо под Windows. Интегрираната среда за програмиране най-често осигурява средства за подготовка, компилация, коригиране на грешки и изпълнение на програмата. Тя предоставя разширен многопрозоречен редактор и обслужва пълния цикъл за създаване и поддържане на изпълняваната програма. Интегрираните среди се използват широко в практиката, тъй като при работа с тях компилирането, свързването и тестването се осъществява, без да се напуска средата на редактора. Интегрираната среда повишава производителността на програмисткия труд. Освен интегрирани среди като допълнителни средства могат да се прилагат мениджъри на проекта, автоматически генератори на приложения и др.

Към инструменталните средства има група други програми. В тях най-често се включват библиотеките от програми, в които се съхраняват обектни модули за някои често използвани процедури, които автоматично се извличат оттам чрез свързващи редактори.

2.3 Приложен софтуер

. Една от причините за огромното разпространение в света на PC- съвместимите персоналните компютри, е софтуера- на първо място удобните за работа операционни системи Windows и пакетите приложни програми, разработени от Microsoft.

Потребителите, неприемащи монопола на Microsoft, могат да изберат друга операционна система, алтернативна на най-използваните в PC ОС Windows– например LINUX.

Едва ли е възможно да се класифицира огромното море на приложния софтуер. Условно има две големи групи:

- пакети приложни програми;
- приложни програмни системи и клиентски софтуер, разработен по заявка.

Пакетите приложни програми представляват специализирани програмни системи със собствено управление и език за общуване с потребителите. Към PC има редица приложни пакети, реализиращи основни компютърни технологии (MS Office, Web-браузъри и e-mail клиенти и др.). Трябва да се отбележи , че независимо от широкото разпространение на пакетите, работещи в Windows платформа, за всеки от тях вече има еквивалентни пакети, работещи под управлението на други ОС.

Към втората група се отнасят всички програми, създадени за решаване задачите на потребителите, и всички приложения, разработени от потребителите и записани в системната памет на компютрите.

Качеството на софтуера е много важно. То се гарантира от спазването на съответните стандарти за софтуер. Основния стандарт, свързан с проектирането на софтуер, е **ISO 9241**. Има европейски и национални стандарти.

При работа с програми архивирането трябва да бъде важна дейност за потребителите. Всяка ОС има службна програма за **архивиране**. Освен тях има разработени архивиращи програми с различна степен на компресия на данните.

Специално внимание трябва да се обърне на **компютърните вируси и антивирусния софтуер**.

Компютърните вируси са специални програми, създадени с цел унищожаването на други програми или файлове с данни за DOS и Windows. Тези програми се отнасят към т.н. **вредно програмно осигуряване (malware)** . Терминът **вирус** е заимстван от биологията. Компютърният вирус е **паразитна програма**, която следва предварително зададени логически инструкции за действие и самосъхранение. Тя е създадена от човек и се пренася чрез всички средства, които се използват за пренос на данни от една компютърна система на друга. Вирусите притежават свойството самовъзпроизвеждане, което е причина за тяхното бързо разпространение. Те се различават по областите на разпространение в компютърната система и поражението на файловете. По-често срещаните са:

Резидентни - разполагат се в паметта и пречат на нормалното функциониране на програми;

Boot- секторни - разполагат се в boot - секторът на дискетите или дисковете, като копират системните файлове на друго място;

Стелт (Stealth) - причиняват най-големи щети, тъй като тяхното присъствие трудно се засича от програмите за антивирусна защита. Те не променят размера на заразения

файл. Секторите на диска, в които те се разполагат се маркират като лоши, въпреки че не са повредени, Освен това те притежават самопроменящ се код, като допълнителна самозащита.

Макро - причиняват правописни и стилистични грешки в текстове и таблици, създадени с текстови редактори и електронни таблици.

Файлови - поразяват изпълними файлове (EXE, COM) и пречат на зареждането на програмния код;

BIOS - поразяват BIOS на компютъра, като променят неговата конфигурация, което прави някои устройства или цялата система неизползваема. Присъствието на вируси най-често е свързано със забавяне работата на компютърната система, появяване на необичайни съобщения за грешки, безпричинно изчезване, повреждане или промяна в размера на файлове, отказ за достъп до дадено устройство, намаляване на количеството памет, блокиране на клавиатурата или мишката!, и настъпването на други странни събития.

Понякога вирусите могат и да са "безобидни"- разкривяване на екранното изображение, звукови ефекти и др. подобия, но това не ги прави по-малко нежелани.

Механизмът на вируса при поразяване има четири основни фази:

- разпознаване- вирусът разпознава дали вече даден файл е заразен от него;
- копиране- вирусът търси "удобна" част от програмата, за да се копира в нея;
- действие- при определени условия (копиране на файл, отваряне на дадено меню и т.н.) се активират вградените в програмата вирус поразяващи функции;
- придвижване- вирусът се връща в програмата, от която се е стартирал, за да може извиканата от потребителя програма, която е поразил да започне работа.

Не е коректно причисляването на компютърните бъгове (bug) и червеите към вирусите. Под **бъг** се разбира грешка в хардуера или софтуера. Тази грешка може да е предумишлена, но не винаги. На тази база се разработват и системите за търсене на грешки и коригирането им (debugging). Подобна е ситуацията и при **компютърните червеи**. Те се разпространяват по мрежа и са самостоятелна или многопрограмна единица (за разлика от вирусите, паразитиращи в среда на друга програма).

Пътищата за заразяване са най-общо следните:

- заразна програма се зарежда и стартира от диск, дискета или др. носител.
- заразяването се извършва от **boot** сектора - (при стартиране на компютъра).
- заразяване по мрежата и електронната поща.
- използване на "пиратското" програмно осигуряване и неговия обмен между потребители.

Наличието на вирус в компютъра не винаги се установява навреме от неговия потребител.

Типичните симптоми за наличие на вирус са: на екрана се появяват неизползваеми текстове, обекти или разбъркани неща; забавена работа на компютъра; продължително време за зареждане на програмите; неочакван недостиг на памет; увеличен обем на програмите и намаляване на дисковото пространство; безпричинно изчезване, промяна или повреждане на файл; съобщения или повреди на дисковите устройства; нарастване броя на повредени сектори по носителите; блокиране на мишка/клавиатура и др.; неочаквани анимации, звук и др.

За защита срещу атаката на компютърните вируси са създадените специализирани програми, т. нар. **антивирусни програми**. Тяхното действие, ако се изхожда от природата на вирусите, е както на системно, така и на потребителско ниво.

Методите, по които може да се класифицира проблематиката на антивирусните изследвания и съответно създавания антивирусен софтуер, са претърпели значително развитие. Те са следвали промените и еволюцията в създаването на компютърните вируси и техните разновидности, като в няколко отделни случая има и изпреварване -

създаване на класификационна група, преди още да се появят съответните вирусни представители. Основните форми на антивирусни програми са следните:

- скенери;
- протектори;
- проверители.

В началото, когато е възниквала необходимостта от антивирусен софтуер и се е поставяло неговото начало, определящи са били идеите на авторите на вируси. Използването на отделен файл за пренасяне и съхраняване на компютърните вируси извежда на преден план функцията преглеждане на съдържанието им и сравняването му с определени уникални последователности, характеризиращи компютърния вирус. Това най-общо е процес на сканиране на съдържание в границите на определени обекти, имащи своите координата в компютърното пространство на компютърната система. Така възниква първата форма на антивирусен софтуер, известна под името **скенери**.

Едновременно с това се стига до идеята за използване на отделни клетки в оперативната памет като среда за скриване на отделни ключови елементи на компютърните вируси при реализация на съответния вирусен сценарий. Имайки информацията за тези характерни клетки и особено за действията около тях в определени времеви интервали, е възможно да се възпрепятства проникването на компютърните вируси и техните разновидности в съответния компютър. При първоначални реализации е имало значение кога се извършват тези проверки за присъствие и евентуално спиране на вирусния код - дали само при първоначалното стартиране на системата, дали само при отделно посочен процес или процедура, или през цялото време на функциониране на системата. Така възниква и втората основна форма на антивирусен софтуер известна под името **протектори**.

С развитието на идеите за защита от вируси, се е стигнало до практическото разбиране, че ако една система е свободна от вируси и друг злонамерен софтуер, тя може да бъде описана чрез определени математически показатели с използване на определени изчислителни ресурси по такъв начин, че става възможно да се открие и да се възпрепятства промяната на един единствен бит, байт, дума и т.н. в компютърните ресурси. Тази защитна концепция, реализираща проверка за интегритета или проверка за целостта, е много близка до абсолютната защита, гарантираща съответна сигурност, но за съжаление тя има твърде висока цена. В близкото минало високи изчислителни разходи, които тя изисква, ограничават използване само там, където има мощен процесор със съответната операционна среда. Така възниква и третата основна форма на антивирусен софтуер, известна под името **проверители**.

Огромно е разнообразието на софтуерни фирмени решения за борба с вирусите с **комбинирани възможности**. Популярни са **Norton AntiVirus, Nod32, AVG, MackAfee, Caspersky Antivirus** и други

В България, специализирано в борбата с вируси, е Националната лаборатория по компютърна вирусология при БАН. Тя предлага комплект от антивирусни програми, част от които са **VirC, F-prot, F-Macrow, F-Stop, Agent** и др. По нейни данни 80% от случаите на заразяване в страната стават чрез комуникационни връзки, а останалите 20% - чрез носители (например дискети).

При съмнение за вирус най-добро решение е рестартиране от бутон **Reset** а не чрез клавишната комбинация за "топъл" рестарт. Следваща стъпка е преформатиране на информационните носители.

Могат да се направят следните **препоръките** за избягване на проблеми с вирусите при работата с компютрите:

- Да не се работи с нелицензиран и непроверен софтуер

- Да се използва защитна стена
- Да се инсталира антивирусна програма, за която редовно да се обновяват вирусните дефиниции
- Редовно да се проверява компютъра за наличие на вируси и или при съмнения за зараза и да бъдат изтрити откритите вируси
- Да се избягва отварянето на непознати изпълними файлове или писма с прикачени файлове
- Да се съхранява важната информация от твърдия диск на системата на дял различен от системния и на други копия (CD, USB RAM и т.н.) за да може да бъде възстановена след тотална зараза.

ТЕСТ

1. Кой от отговорите не се отнася до предназначението на операционната система?
 - а) съвкупност от команди за управление в Интернет;
 - б) управлява процесора, паметта, периферните устройства и данните в изчислителната система;
 - в) посредник между потребителя и изчислителната система;
 - г) управлява хардуера и софтуера и осигурява интерфейс с потребителя.
2. Посочете грешния отговор:
 - а) WINDOWS е многопроцесорна операционна система;
 - б) WINDOWS е многозадачна операционна система;
 - в) WINDOWS е многопроцесна операционна система;
 - г) WINDOWS е операционна система с графичен интерфейс.
3. Машинният език зависи от:
 - а) типа на микропроцесора;
 - б) конфигурация на компютъра;
 - в) типа на вътрешната памет;
 - г) типа на външната памет.
4. От какъв тип е програмата за изчисляване на дипломите за више образование?
 - а) пакет приложен софтуер;
 - б) операционна система;
 - в) приложен софтуер на клиент;
 - г) транслятор.
5. Кое не е елемент на базовото програмно осигуряване?
 - а) езиците за програмиране;
 - б) операционните системи;
 - в) антивирусните програми;
 - г) интегрираните среди за програмиране.

6. От програма на какъв език е този ред- **add ax, bx** ?
- а) машинен;
 - б) C++ за MAC;
 - в) Асемблер за Интел ;
 - г) EBN Pascal.
7. От програма на какъв език е този ред- **00101011 11000001**?
- а) машинен ;
 - б) C++ за MAC;
 - в) Асемблер за Интел ;
 - г) EBN Pascal
8. Кой вид софтуер е “зловреден”?
- а) скенер;
 - б) протектор;
 - в) проверител;
 - г) заден люк.
9. **Дебъгер** е :
- а) програма за тестване и настройка на клиентска програма;
 - б) програма за за превеждане на изходната програма в машинен код;
 - в) антивирусна програма;
 - г) програма за електронна таблица.
10. **Фърмуер** е:
- а) програми във ВЗУ;
 - б) програми в RAM;
 - в) програми в ROM;
 - г) програми в диск.

. Глава 3. КОМПЮТЪРНИ МРЕЖИ И ИНТЕРНЕТ

3.1 Основни понятия от телекомуникациите

Предаването на информация на големи разстояния се осъществява от електронните телекомуникационни системи (телефон, радио, телевизия, и др.) във вид на съобщения (текст, глас, изображение, данни). Съобщението се предава с помоща на сигнали. **Сигналите** са изменящи се физически величини (ток, електромагнитно поле, светлина, звук), изобразяващи съобщението. Сигналят еднозначно съответства на предаваното съобщение.

Съществуват два вида сигнали:

- аналогови (непрекъснати);
- цифрови (дискретни).

Аналоговият сигнал има безкраен брой значения в ограничен интервал от време. Примери за източници на аналогови сигнали са човешката реч, старите грамофонни записи, компактдискете със звукозаписи, радиото, телефона. При говор, във въздуха се образуват акустични вълни. Колебанията на зоните с високо и ниско налягане при

това са аналогови сигнали. Микрофоните преобразуват физическите колебания на въздуха в електрически аналогови сигнали.

Цифровият сигнал за определен интервал от време има краен брой значения на амплитудата..

Всеки сигнал има три важни характеристики- мощност, честота и продължителност. Аналоговите периодични сигнали се характеризират с амплитуда (A), честота (f), и фаза. Човешкото ухо различава най-добре звуци в диапазона 40 Hz - 18000 Hz.

Разликата между най-високата (f_v) и най-ниската честота (f_n), присъстващи в сигнала, се нарича честотна лента (честотен спектър) на сигнала:

$$F_c = f_v - f_n$$

Ширината на спектъра се нарича основна честотна лента. Сигналите от типа на реч, музика и видео имат сложен честотен спектър. Тези спектри могат да бъдат представени чрез графика на зависимостта на амплитудата от честотата.

За да се предаде едно съобщение на разстояние е необходима **комуникационна система (КС)**. Тя включва съвкупността от технически средства, необходими за предаване на съобщения от източника към получателя (източник на съобщения, предавател, комуникационна линия, приемник и получател на съобщенията) , програмни средства и правила за функциониране. КС е система за предаване на съобщения, а не на електрическа енергия.

Според вида на предаваните съобщения, КС са телефонни (гласови съобщения), телексни и телеграфни (текстови съобщения), факсови, факсимилни (за предаване на неподвижни изображения), телевизионни и видеотелефонни (видео-изображение), системи за предаване на данни и телеизползватели (за телематични съобщения-данни за измервания от разстояние).

Общите характеристики на КС се определят и ограничават от индивидуалните параметри на предавателя, комуникационната среда и приемника. В тези системи се предават и аналогови, и цифрови сигнали.

Съобщението има количествена характеристика – **количество информация**– измерва с в битове (bit).

Количеството информация, което може да предаде комуникационната система за единица време определя нейната **пропускателна способност**. Единицата за нейното измерване е **бит за секунда (b/s)**, тя се използва за измерване на скоростта на предаване на данни (информация) по комуникационните линии и в компютърните мрежи.

При цифровите сигнали двоичните данни се предават чрез кодиране на всеки бит от данните в сигнални елементи.

Преобразуването на съобщението в сигнал е свързано с реализиране на три операции:

- **преобразуване** на съобщенията от неелектрическа величина в електрическа;
- **кодиране** – построяване на сигнали по определен принцип и правила;
- **модулация** – въздействие върху някой параметър на електрическия сигнал – амплитуда, честота и фаза.

Кодирането има три разновидности:

а) първично кодиране, когато построяването на сигнала става според определен математически израз (например преобразуване на текст в ASCII или UNICODE - код).

б) вторично (шумоустойчиво) кодиране, при което към основното съобщение чрез избран метод се добавят допълнителна (служебна) информация, чрез която се откриват

и коригират евентуални грешки в съобщението, станали по време на неговото предаване по комуникационните линии ;

в) кодиране в комуникационната линия (линийно кодиране) - преобразуване на първичния сигнал във вид, удобен за по-ефективно предаване по линията.

В комуникационните системи по време на процеса на предаване могат да възникнат смущения, които да развалят съобщението. Всяко подобно нежелано смущение в системата се нарича "шум". Всяка комуникационна система се характеризира с величината "**съотношение сигнал/шум**". Колкото по-голяма е неговата стойност, толкова по-добре се дешифрира полезния сигнал. Практическият критерий за ефективна комуникация е правилно разбраното съобщение след приемането му.

Комуникационна линия (среда, медия) се нарича физическата среда, която се използва за предаването на сигналите от предавателя към приемника.

Комуникационни линии са:

- а) специални кабели;
- б) телефонни линии;
- в) наземни линии;
- г) линии за спътникова връзка;
- д) глобалните и локални мрежи;
- е) безжични линии;
- ж) микровълнови линии;
- з) стандартни радио връзки.

Между предавателя и приемника в КС може да има много линии (проводници), по които данните да се предават паралелно (до няколко десетки бита едновременно). Този тип предаване се използва обикновено за много къси разстояния (до няколко десетки метра), и се нарича **паралелно предаване**. То е много скъпо и ненадежно на големи разстояния. За големи разстояния се използва само една линия за предаване, по която предаването става бит след бит, последователно във времето. Този тип предаване се нарича **последователно (серийно) предаване**.

Има две категории комуникационни среди- управляеми и неуправяеми.

В **управляемата** среда сигналите се насочват по физическия път, такива са усуканите двойки медни проводници на телефонните линии, коаксиалните и оптичните кабели. В **неуправляемата** среда сигналите се разпространяват през въздуха, водата и безвъздушното пространство. В тази среда предаването и приемането стават чрез антени

Комуникационната система може да включва повече от един тип от изброените комуникационни линии. Всяка от тях има различни характеристики, което се отразява на общата производителност на комуникационната система.

Основните типове данни, които се предават са телефонни, компютърни данни, видео и изображения (картини, фотографии). Тези типове данни се разделят на две категории – аналогови и цифрови. Всяка от тях има различни изисквания към КС.

Комуникационен канал е съвкупност от средства, осигуряващи предаване на сигнал от една точка на комуникационната система до друга нейна точка. Комуникационният канал в компютърните комуникации е логическо понятие.

При предаването по каналите се наблюдава закъснение, затихване и деформиране на сигналите. Затихването се измерва в **децибели (db)**.

Всеки цифров сигнал има определена **честотна лента**. При опит за предаване на такъв сигнал чрез определена комуникационна среда, нейната природа ограничава честотния спектър на сигнала който може да бъде предаван. За всеки канал важи правилото- колкото е по-широка честотната лента на сигнала предаван по-него, толкова

по-скъпо е предаването. Изхождайки от практически и икономически причини е прието цифровата информация да се представя чрез сигнали с ограничен честотен спектър. Ограничаването на честотната лента обаче предизвиква деформации в сигнала, което затруднява правилното интерпретиране на информацията в приемника – колкото е по-ограничена честотната лента толкова е по-голяма вероятността за грешки в приемника.

Каналите могат да осъществяват предаване на сигнали в един и същ момент от време по следните начини:

- само еднопосочно – това се нарича **симплексен** режим на работа ;
- двупосочно – това е дуплексен режим на работа (нарича се още **пълнен дуплекс**);

Възможен е и трети режим на работа на канала, когато сигнал от предавателя към приемника се подава в един интервал от време, а обратно – в друг интервал от време. Този режим се нарича **полудуплексен**.

Телефонните канали осигуряват дуплексен режим на предаването на разговори, докато радиотакситата си обменят информация в полудуплексен режим.

Комуникационната система е многоканална, ако осигурява няколко паралелни канала за предаване на различни съобщения по една обща комуникационна линия. При това е необходимо да се използва уплътнение (мултиплексиране) на линията. Отделните съобщения се преобразуват в електрически сигнали, които след това се смесват от устройството за уплътнение (мултиплексор) в един общ групов сигнал .

За разделяне на отделните подканални е необходимо те да се отличават по някакъв признак. При многоканалните комуникации се използват няколко способа за разделяне на канали (мултиплексиране): времеделение (TDM – Time Division Multiplexing), честотно деление (FDM – Frequency Division Multiplexing) статистическо деление (SM – Statistical Multiplexing). При честотното деление (FDM) на канали (сигнали), лентата на пропускане на линията (груповия канал) се разделя на m равни части (m подканала) за едновременно предаване на m сигнала, всеки от които се модулира за да попадне в честотната лента на пропускане на съответния подканал. В системите с времеделение (TDM) всеки подканал разполага със свой периодично повтарящ се интервал от време, не заема от друг подканал (сигнал), а честотните спектри на всички подканални (сигнали) лежат в една и съща честотна област.

Под **комуникационна мрежа** (КМ) се разбира съвкупност от комуникиращи устройства, свързани помежду си чрез комуникационни линии. Комуникационната подмрежа е съвкупност от комуникационни линии и свързаните чрез тях устройства, наречени междинни мрежови възли. Последните извършват две основни функции- **комутация и маршрутизация**. Пренасочването на съобщение от един вход на междинния мрежов възел към определен негов изход се нарича комутация, а намиране на оптимален маршрут за преминаване на съобщението през комуникационната помрежа- маршрутизация.

3.2 Компютърни мрежи и стандарти

Компютърната мрежа е съвкупност от компютри, свързани помежду си чрез линии за комуникация с цел обмен на информация между тях. Компютърните мрежи, в които крайните възли са компютри, представляват частен случай на комуникационните мрежи.

В зависимост от географското разположение на компютрите, компютърните мрежи се подразделят на:

- **глобални компютърни мрежи** (английското съкращение за тях е WAN дават възможност за свързване на компютри по цялото земно кълбо, най-популярната (но не единствена) глобална мрежа е Internet;

- **локални компютърни мрежи** (ЛКМ, на английски- LAN)- обединяват компютри в рамките на една сграда, лаборатория, компютърен клас, обикновено разстоянието между крайните компютри е до 2 км;

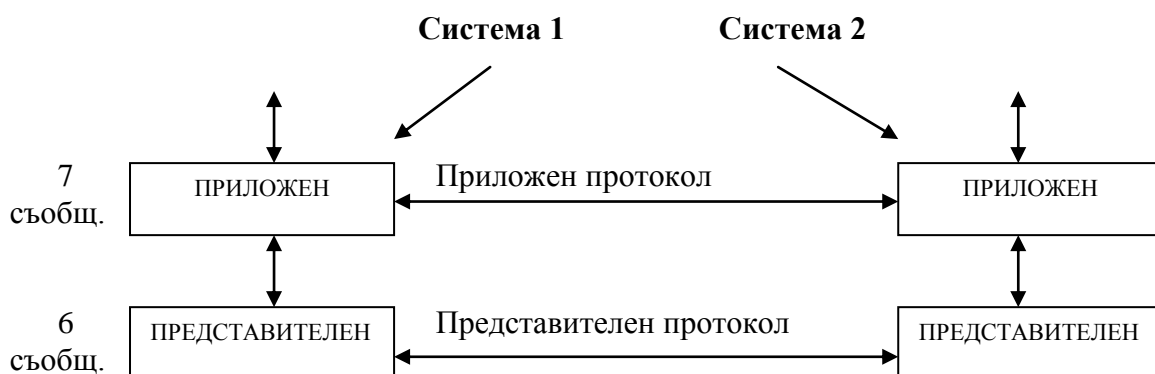
- между тези две крайни по отношение на разстоянията между свързаните в тях компютри и броя на свързаните в тях компютърни мрежи, съществуват **регионални (MAN)** и други видове мрежи.

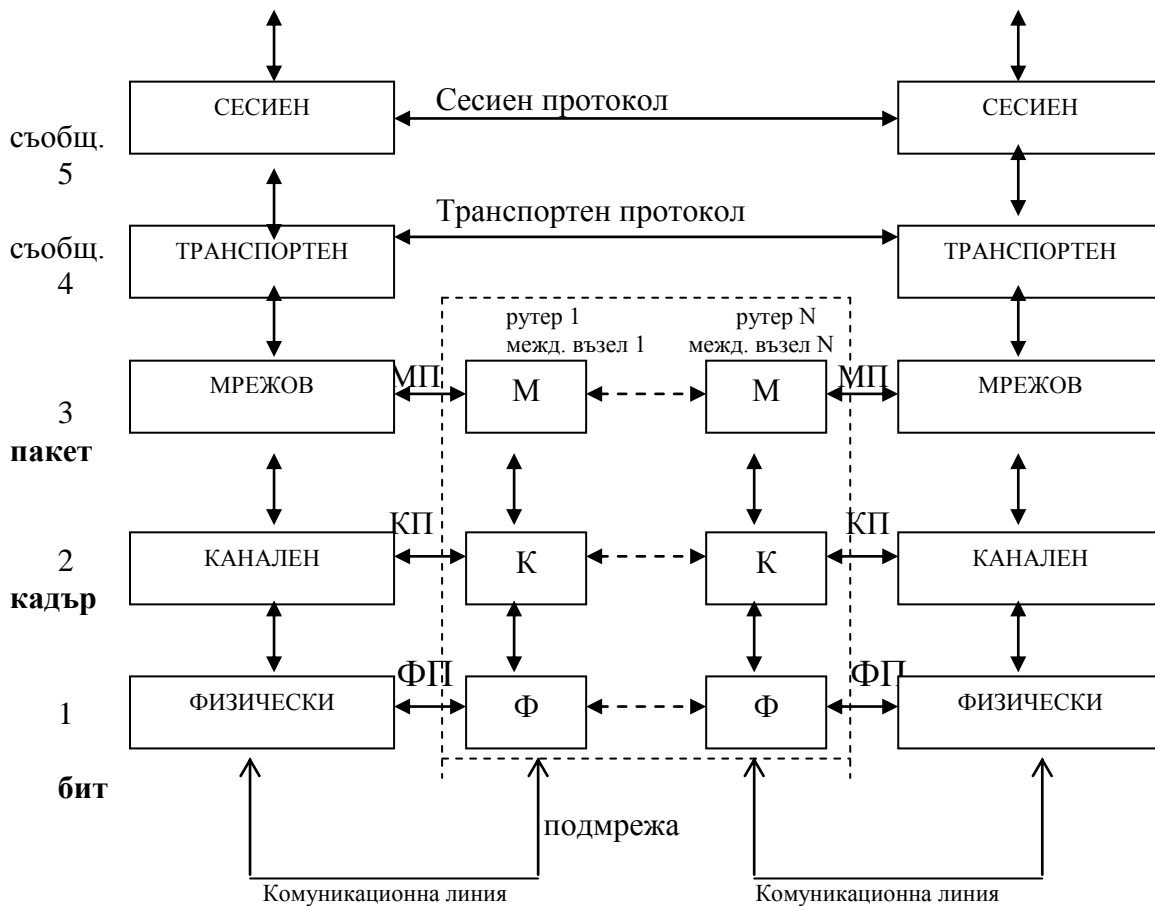
Различните мрежи използват някои от следните **методи за комутация** - комутация на канали, комутация на съобщения, комутация на пакети в два режима- дейтаграмен и режим на виртуално (логическо) съединение, и бърза комутация на пакети.

За уеднаквяване на подходите и стандартите при изграждане на компютърни мрежи, международната организация по стандартите (ISO) и международния комуникационен съюз (ITU) са разработили модел, наречен еталонен модел за свързване на отворени системи- **OSI**.

Основната идея на модела е разделянето на комуникационните функции на всяка една система на слоеве. Според модела взаимодействието между две потребителски системи се осъществява чрез взаимодействие между едноименните слоеве. Правилата, по които се осъществява това взаимодействие, се наричат комуникационни протоколи.

Комуникационният OSI-модел се състои от 7 слоя (по тази причина понякога го наричат "седемслоен модел")- физически, канален, мрежов, транспортен, сесиен, представителен и приложен. Всеки слой изпълнява определена логическа функция. Тези формулирани от модела функции улесняват разработването на нови стандарти. На фиг.4.1 е показано разположението на тези слоеве спрямо физическата линия и потребителя на информацията.





Фиг. 3.1

Физическият слой е свързан с физическото пренасяне на информацията. Тук се определят електрическите и механични връзки между мрежовата кабелна система и функциите, осъществяващи предаването на битове от едно устройство към друго. В този слой се описват и кабелните спецификации (например 10Base2, 10BaseT). На това ниво се специфицира интерфейса към горния слой.

Каналният слой контролира пренасянето на данните. Определят се стратегията и механизма за достъп до общата съобщителна среда (откриване на конфликти, описват се и се формират форматите за предаване). Определя се дали има грешки при предаване на данните и, ако има, те се отстраняват.

Мрежовият слой избира оптимален път, по който съобщението ще пътува към определената станция (това се нарича маршрутизиране). По-горните слоеве не се занимават с маршрутизиране.

Транспортният слой обработва възникнали грешки (например изгубени пакети), тук се извършва фрагментиране на по-големите съобщения на по-малки (става разбиването на отделни пакети).

Сесийният слой осигурява поддържането на диалога между възлите и работещите приложения.

Представителният слой "представя" информацията на взаимодействащите помежду си обекти взети от приложния слой. Съобщенията от приложния слой се преобразуват в удобна за пренасяне форма.

Приложният слой е "прозореца", чрез който потребителите на системи обменят помежду си информация.

Предназначението на свързочната среда е да транспортира потока от битове от мрежовите карти на компютрите-предаватели към тези на приемниците.

Горните слоеве на ЛКМ (от мрежовия слой на модела OSI нагоре), се реализират софтуерно от мрежовите операционни системи (МОС) и приложенията, които са инсталирани на компютрите в мрежата.

Физическите съединения могат да осигуряват както последователно (побитово) предаване, така и паралелно (побайтово) предаване на данни. Физическите съединения могат да са постоянни (наети линии) или временни (комутируеми) линии.

OSI – моделът е изграден на три основни понятия: услуга; интерфейс; протокол. **Услугата** е логическа функция по осигуряването на съседния слой на OSI в един възел на мрежата. **Протоколът** е съвкупност от правила за взаимодействие на обекти от едноименни слоеве за различни възли от мрежата. **Интерфейс** са правилата за взаимодействие на обектите от съседните слоеве на един и същ възел. **Спецификацията** на протоколите се заключава в определяне на формата на неговите полета за данни **PDU** (Protocol Data Unit), семантиката на полетата на тази единица, допустимата последователност на предаване на отделните протоколни единици. Дефинирането на **услугите** е свързан с определяне на какви услуги да се предоставят на горния слой, но не и как това да стане. **Адресацията** се състои в указание за обектите на по-горния слой на точките SAP (Service Access Point), чрез които те могат да получат достъп до услугите на по-горния слой.

Обектите от слой N си взаимодействат помежду си чрез съединения с обекти от слой N-1, по който се предават масиви от информация, наречени **Протоколни единици за данни на слой N**. Всеки слой N-1 осигурява услуги за обекти от слой N. На всеки обект може да му се предоставят данни по съединения тип "ТОЧКА – ТОЧКА" или "ТОЧКА – МНОГО ТОЧКИ". Използват се три типа съединения: симплексни; полудуплексни; дуплексни.

Протоколите в отворения модел извършват следните основни функции: Фрагментация и дефрагментация; Капсулация; Управление на съединението; Доставка на PDU в правилен ред; Управление на потока от данни; Контрол на грешките; Адресация; Мултиплексиране; Услуги на предаването.

Фрагментацията е процесът по разделянето на потока данни на части (блокове, фрагментация, сегментация). Блоковете се наричат протоколни единици от данни (**PDU** – Protocol Data unit). PDU има фиксирана и променлива дължина за едни или други протоколи. Използват се следните стандарти:

- ATM стандартът има PDU - 53 байта;
- Ethernet е с променлива дължина до 1526 байта;
- Frame Relay е с променлива дължина до 1608 байта;
- FDDI е с променлива дължина до 4500 байта;
- X.25 е с променлива дължина до 4096 байта или 128 б (с подразбиране).

В приемната точка данните се възстановяват в първоначалното съобщение, което се нарича **дефрагментиране**.

Капсулацията е процес, при който PDU от по-горен слой се вмъква в полето "данни" на PDU от по-нисък слой и към нея се допълва служебна информация за предаването. Служебната информация се състои от:

- адрес на управляващата и контролната информация;
- адресна информация за правилна доставка на PDU до нужния адресат;
- управляваща информация – за правилното функциониране на протокола;

- контролна информация – за откриване или коригиране на възникнали грешки при предаването на PDU.

Комуникационният сценарий на отворения модел за комуникации може да се представи чрез предаване на съобщения между крайните възли 1 и 2 на една компютърна мрежа. Съгласно сценария, процес А от система 1 предава съобщение на процес В в система 2. За целта А подготвя масив от данни и призовава приложния слой, който установява логическа връзка с приложния слой от система 2 чрез приложен протокол. Протоколът изисква услуги от представителните слоеве на системи 1 и 2, като те използват за връзка съответен протокол. В слой 6 на 1 съобщението идва от слой 7 във вид удобен за транспортиране. Преобразува се кодът, азбуката, съгласува се формата и съобщението се предава към сесийния слой 5.

В слой 5 се организира предаване на съобщението, като се установява съединение между процеси А и В. Използва се сесиен протокол за поддръжка на съединението, идентификация на съединението, управление на диалога и разпадане на съединението.

В слой 4 съобщението се транспортира по комуникационната подмрежа от край до край. За целта се организира двупосочно съединение и състоянието му се контролира от транспортен протокол.

В слой 3 (мрежов) се организира съединение между съседните мрежови възли, извършва се маршрутизация, комутация, управление на натоварването. Съобщенията се накъсват и предават във вид на отделни пакети.

В каналния слой се организират канали за предаване на данни между два съседни възли. Информацията се предава във вид на блокове – наричани „кадри”. Извършва се синхронизация по кадри, контрол на грешките, управление на потока от данни, контрол на състоянието на канала и др.

Във физическия слой се използва комуникационна (физическа) среда за предаване на неструктуриран поток от битове. По линията може да има уплътнение и едновременно могат да се предават няколко съобщения. Извършва се и контрол на линията и синхронизация по битове.

Между равностойните слоеве на комуникационните системи няма физическа връзка с изключение на физическите слоеве.

Във всеки слой N на предаващата система към протоколната единица (PDU), получена от слой N + 1, се добавя собствена заглавна част (header). Изключение прави каналният слой, който добавя не само заглавна част (header), но и крайна част (trailer).

В приемната система 2 се извършват обратни преобразувания за възстановяване на съобщението. Всеки слой на модела може допълнително да фрагментира PDU на горния слой, за да удовлетвори своите изисквания за дължината на предавания блок от данни.

Преобразуването на съобщенията в сигнали е функция на физическия слой. Тя е свързана с необходимостта от преобразуване на предаваните съобщения в определен вид сигнали, съответстващи на изискванията на използваната физическа среда за предаване. Съществуват четири възможности, съответно за преобразуване на:

- аналогово съобщение в цифров сигнал;
- цифрово съобщение в цифров сигнал;
- цифрово съобщение в аналогов сигнал;
- аналогово съобщение в аналогов сигнал.

Преобразуването на аналоговите сигнали в цифрови е свързано с влиянието на шумовете върху елементите на комуникационните системи. В аналоговите системи заедно с полезните сигнали се усилват и шумовете от различни източници и като резултат се влошава съотношението "сигнал/шум" и качеството на приемане. В

цифровите системи се използват само две състояния, представящи 0 и 1, и в тези системи практически **отсъстват шумове** при предаване на сигнали. Това е основната причина за по-голямата перспектива на цифровите системи.

Сред преобразуванията на аналоговите съобщения в цифрови сигнали най-голямо практическо приложение има импулсно-кодната модулация – **ИКМ** (Pulse Code Modulation - PCM). ИКМ е процес, при който дадено аналогово съобщение се дискретизира по време, след което дискретите му се квантуват по ниво и чрез кодиране се преобразуват в цифров сигнал.

В компютърните мрежи най-много се използват управляемите среди – кабели с усукани двойки проводници, коаксиални кабели и влакнесто-оптични кабели. Използват се и неуправляеми среди – радиовълни и лазерни лъчи през въздуха. Всяка от тези среди има своя ниша в смисъла на честотната лента (колко данни може да се пуснат по нея), закъснение, цена и простота на инсталирането и поддръжката.

Влакнесто-оптичните кабели се състоят от отделни оптични влакна (модове), по които сигналите се предават под формата на модулирани светлинни импулси. Тези кабели са с най-висока устойчивост към електромагнитните смущения, покриват най-големи разстояния и осигуряват най-високи скорости на предаване, но те все още най-скъпи. Заради високите скорости на предаване се използват повече в опорните мрежи, но навлизат и в ЛКМ. През всеки 30 км се поставят повторители, за разлика от тези с усукани двойки – през 5 км, което намалява общите разходи.

Безжичните комуникации се използват, когато потребителите са в движение или там, където е трудно да се прекарат кабели. Използват се три основни способа за безжично свързване на компютри – технологията SST, тяснолентовата високочестотна радиовръзка и инфрачервената технология. Технологията SST използва радиочестотния диапазон от няколко стотин мегагерца до около 60 GHz. Типичните SST безжични ЛКМ работят със скорост около 2 Mb/s на дистанции до 200 метра. Една тяхна алтернатива са маломощните високочестотни безжични ЛКМ, с работна скорост 15 – 20 Mb/s, работещи на ограничени разстояния – от 10 до 50 метра.

Инфрачервената безжична връзка е удобно да се използва в лабораторни условия на разстояния до 30 метра.

3.3 Локални компютърни мрежи

Локалните компютърни мрежи (ЛКМ, английското съкращение е LAN) са най-големия брой мрежи, получили разпространение в предприятия, фирми, банки, университети, училища и много други места, където има нужда от съвместна работа на няколко десетки компютъра. **ЛКМ** се нарича високоскоростна система за предаване на информация и ползване на общи мрежови ресурси (процесорно време, памет, файлове и входно-изходни устройства) между компютри, принадлежащи обикновено на една организация и разположени в компактен географски район (до 2 км).

Разликата в географско отношение по сравнение с глобалните мрежи води до различни технически решения в ЛКМ. Обикновено тези мрежи са собственост на организацията, която притежава компютрите, сървърите, принтерите и др. устройства, свързани към мрежата. За глобалните мрежи това не е така. Поради това правилния избор на ЛКМ е важен за размера на капиталните вложения за изграждането и поддръжката на мрежата. Скоростта на данните вътре в мрежата е значително по-голяма от тази в глобалните мрежи.

Според функциите, които изпълняват, компютрите в една локална мрежа се делят на работни **станции (клиенти)** и **сървъри**. Освен това в тях с цел свързване и разширение може да има и други устройства - повторители, концентратори, мостове, маршрутизатори и шлюзове, които се наричат междинни мрежови възли.

Работната станция е потребителски компютър, чрез който се получава достъп до мрежовите ресурси. **Сървърът** е компютър, който предоставя мрежови ресурси и услуги, както и компютър, обслужващ и управляващ мрежата. В една локална мрежа могат да бъдат конфигурирани един или повече сървъра. Не е задължително под понятието "сървър" да се разбира отделен компютър, това може да бъде и софтуерен продукт, стартиран върху компютър (WEB сървър, e-mail сървър и т.н.).

Освен компютрите, в състава на локалната компютърна мрежа има:

- а) съобщителна среда (проводници, ефир и др.);
- б) хардуерен интерфейс между съобщителната среда и работната станция - наречен мрежов адаптер;
- в) вграден в мрежовите адаптери системен софтуер (фърмуер), който реализира първия и втория слой на OSI модела;
- г) съвкупност от протоколи и услуги от по-високо ниво, които осигуряват надеждно обслужване;

д) софтуер, позволяващ на потребителя достъп до локална мрежа;

Последните два компонента образуват заедно **мрежовата операционна система (MOC)** и реализират функциите на модела OSI, съответстващи на слоевете 3-7.

За разлика от модела OSI, каналния слой на ЛКМ е разделен на два подслоя - горен подслой (към мрежовия) за управление на логическия канал (LLC), и долен подслой (към физическия) за управление на достъпа към общата комуникационна среда (MAC). Функциите на каналния слой се реализират от мрежовата карта (нарича се още мрежова интерфейсна платка, мрежов адаптер), която е монтирана на всеки възел (компютър), свързан към мрежата.

Класификацията на локалните мрежи може да бъде направена по няколко признака - топология на мрежата, използвани комуникационни линии, способ за използване и достъп до съобщителната среда, услуги и протоколи, предоставени на потребителите. Тези признаци определят в основни линии цената и производителността на мрежите, както и приложенията, които те могат да поддържат.

Съществено значение за локалните мрежи имат комуникационните линии. Това са основно кабелни, но има и безжични мрежи. За класификационната принадлежност на мрежата има значение и броя на едновременно предаваните сигнали по комуникационната линия. Според това има широколентови (broadband) и теснолентови (baseband) мрежи. При широколентовите по една линия чрез метода FDM се организират повече подканални за предаване на много сигнали на различни честоти. В теснолентовите по един канал се предава един сигнал, като правило на честотата на компютъра.

Топология на една мрежа се нарича начинът на разполагането и свързването на отделните и възли. Има следните основни топологии за ЛКМ:

- дървовидна - към "корена на дървото" са свързани на различни нива мрежовите възли, образуващи дървовидна структура;
- шина - към една шина за данни са свързани паралелно мрежовите възли (тази топология може да се счита за вариант на дървовидната) ;
- звезда - крайните мрежови възли са свързани към централен възел във вид на звезда
- кръгова - възлите на мрежата са свързани в кръг ;

Използват се и техни комбинации.

При кръговата топология предаването на данните (за краткост "трафик") се извършва само в едно направление (по посока на часовниковата стрелка) и съобщенията циркулират по мрежата. Всяка работна станция в пръстена получава данни от съседната в кръга и ги препредава към следващата. Данните се предават във

вид на пакети, наречени **кадри**, с фиксирана или променлива дължина (в зависимост от типа на пръстена). Всеки кадър съдържа полезна информация и служебни данни, в които има адреса на получателя. Всички станции проверяват при приемането на кадъра адреса на получателя, и ако той не е техния, те го препредават по кръга. Ако дадена станция разпознае "своя" адрес, тя копира кадъра в своя буфер. По едно и също време в кръга е възможно да има едно или повече съобщения. Първият компютър, включил се в кръга, става мониторна станция за мрежовата дейност, следи за грешки в пръстена и коригира повредите, например загубата на маркера. За кръгова топология има няколко стандарта - FDDI, Token ring, Slotted ring и др. Кръговата топология има **следните предимства**- покрива големи разстояния, кабелите между отделните възли не са дълги и в мрежата могат да се използват оптични кабели. Като недостатък може да се отбележи, че повреда в един от възлите (компютрите) може да доведе до разпадане на мрежата. Освен това в мрежата трудно се откриват повредите.

Шинната топология се характеризира с това, че тук няма повторители или превключватели. Работните станции се включват към линията за предаване (шината) чрез интерфейсни блокове, наречени "тап". Всяко предаване от коя да е станция се разпространява и в двете посоки по шината. Краят на шината е терминиран така, че да абсорбира сигналите, като по този начин ги премахва от нея. Само една станция може да предава в даден момент от време. Предаването се извършва чрез **кадри**. Не трябва да се вземат специални мерки за премахване на кадрите от мрежата, тъй като те се "поглъщат" от терминаторите в края на шината. Предимствата на тази топология са лесното разширение на мрежата чрез безпроблемно добавяне на нови възли, и високата надеждност на мрежата при отпадане на някой от компютрите. Основните недостатъци на тази топология са слабата защита на информацията, затруднената диагностика на системата и ограниченото разстояние, което може да покрие. Примери за ЛКМ с такава топология са безжичните мрежи и вариантите на Ethernet -10Base5 и 10Base2.

При топологията "звезда" има централен комутиращ възел, към който са свързани всички останали възли. Има пасивно и активно свързване в звезда. При пасивното свързване входните сигнали просто преминават в изходящите линии, докато при активното свързващия възел работи като поворител, чете битовете, които влизат в него, автоматично ги регенерира и ги повтаря, изпращайки ги по всички изходящи линии. Предимства на топологията са лесната поддръжка и добавяне на нови възли, улесненото централизирано управление и диагностика на функционирането на мрежата. Основният недостатък се състои в това, че при повреда на централният възел се разпада цялата мрежа.

Съвременните ЛКМ използват един от двата метода за достъп към общата среда (наричат се още протоколи за достъп):

- метод за множествен достъп с откриване на носещата и разпознаване на конфликта- МДОН/ПК, известен е с английското съкращение CSMA/CD;
- Token ring (може да се преведе като маркерен кръг).

Форматът и подреждането на отделните полета в кадрите на двата протокола са различни.

MAC-протоколът CSMA/CD се използва в безжичните мрежи и ЛКМ Ethernet с шинна и звездообразна топологии. Терминът "шина" тук означава високоскоростна линия с ограничена дължина. Дължината може да бъде увеличена чрез допълнителни устройства - повторители. Ethernet първоначално е внедрен от компанията Xerox, сега това е световен и най-използвания стандарт за ЛКМ. Той допуска, че всички станции в мрежата са равноправни и се състезават за заемане на шината с цел предаване на данни. Състезателният протокол работи най-общо по следния начин. Когато в даден момент от време една или повече работни станции искат да се включат към единствения

логически канал, за да поставят в него своите кадри, те "прослушват" канала, за да открият дали някой вече не предава по него. Ако една станция не открие носеща честота в канала, това означава, че той е свободен, тогава тя формира кадър и го излъчва по линията, по която той се разпространява в двете посоки на шината. Мрежовата карта "слуша" канала, докато се разпространява нейното съобщение. Поради крайната скорост на разпространяването на сигнала по шината, е възможно друга станция, която не е открила носеща в канала, да излъчи кадър, който след време ще се "сблъска" с вече излъчен такъв от друга станция. И двата ще се деформират. Станцията, която първа открие този факт на конфликт, изпраща специален заглушаващ сигнал. Всички станции прекратяват предаването, изчакват различен интервал от време (зададен от генератор на случайни числа) и отново изпращат кадрите си. Ако по време на разпространението на кадъра не възникне конфликт, то продължава успешно до края и канала се освобождава. При големи натоварвания на мрежата (много опити на станции да заемат линията) могат да възникнат повече конфликти, и това води до закъснение на кадрите. Рязко спада времето за отговор на мрежата (това е времето от изпращане на заявка - кадър и получаването на данните).

Протоколът Token ring дава по-добро управление върху достъпа в мрежата, отколкото Ethernet . За да вкара данни в кръга, предаващата станция трябва да завладее свободен маркер. Маркерът обикаля по кръга, той може да бъде "свободен" или "зает". Когато е зает, това се отбелязва в съответното му поле. Станцията, желаеща да предава, може да използва свободен маркер, променя състоянието му в "зает", прикрепя своето съобщение към него и го връща обратно в кръга. Всички останали станции разбират, че съобщението е формиран пакет с данни и те не могат да захванат маркера. Компютър-получател приема съобщението, копира данните, установява бита за потвърждение в полето на маркера в 1, за да покаже, че съобщението е прието, и го връща в кръга. Съобщението обикаля по кръга, достига до изпратилата го станция, която преценява дали то е било правилно прието от получателя. Ако е правилно, предаващата станция трябва да освободи маркера и да не го използва отново докато той не мине повторно до нея по кръга.

Ако компютърът, който изпраща кадъра, се повреди преди да е освободил маркера или преди да го въведе обратно в кръга, маркера ще се изгуби, ако не се вземат мерки. За правилното функциониране на мрежата, един от компютрите се определя за монитор на маркера. Той премахва кадри, които са оставени да циркулират многократно по кръга, и ако след определено време маркера не се върне, монитора на маркера генерира нов.

Протоколът Token ring има няколко предимства пред CSMA/CD. Най-напред това е значително по-добрата му производителност. Мрежа Token ring със скорост 4 Mb/s има същата ефективна производителност, както 10 Mb/s- мрежа Ethernet.

Като цяло, състезателния метод CSMA/CD работи по-добре в мрежи, които нямат интензивно използване. В този случай всеки компютър може да предава без да се налага да чака дълго, и вероятността за конфликт е малка. При мрежи с висока натовареност Token ring работи по-добре. Практиката показва, че при тежко натоварване скоростта на данните в Ethernet-мрежа пада от 10 на 5 Mb/s, докато в мрежа Token ring с 16 Mb/s тази скорост се запазва и при тежко натоварване. Token ring е три пъти по-бърза от Ethernet при големи натоварвания. Когато трафикът не е тежък, Token ring е по-бавна, тъй като маркера трябва да се освобождава след всяко използване. Обаче времето за отговор не се увеличава с увеличаване на натоварването. Освен това при този протокол никой от възлите не може да монополизира мрежата, защото времето за предаване е ограничено.

За построяването на локални мрежи са създадени международни стандарти. Международните стандарти за физическия и канален слоеве на LAN са създадени от IEEE и имат означение IEEE 802.X: IEEE 802.1 – съдържа общи постановки и обяснения; IEEE 802.2 описва горния LLC –подслой на каналния слой; IEEE 802.3, 802.4, 802.5, 802.12 описват различни типове LAN. Само IEEE 802.6 описва регионалната (MAN) мрежа. Всички стандарти покриват едновременно физическия и MAC – подслоя на каналния слой.

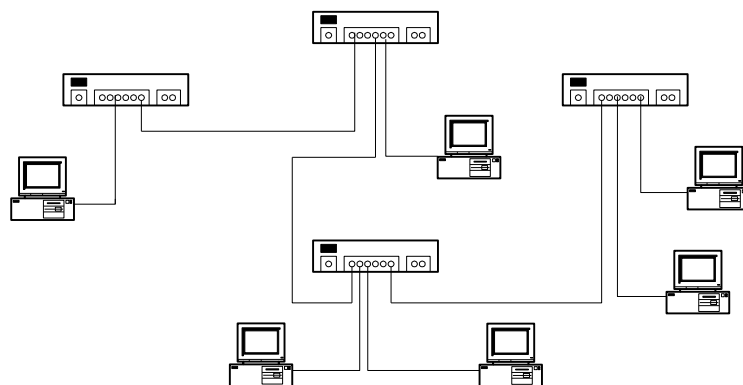
Най-разпространен е стандартът IEEE 802.3, който обхваща изцяло създадения преди него стандарт Ethernet. Този стандарт включва пет версии за мрежи със скорост на предаване на данните 10 Mb/s (10Base5, 10 Base2, 10BaseT, 10BaseF, 10Broad36) и две високоскоростни версии - "бърз Ethernet" (Fast Ethernet) и "гигабитов Ethernet" (Gigabit Ethernet). Този стандарт реализира метод за достъп с откриване на носещата - МДОН/ПК (CSMA/CD). Цифровите сигнали се предават директно в основната честотна лента (baseband). Общата среда може да бъде коаксиален кабел, усукани двойки проводници или оптичен кабел. Скоростта на предаване е 10 Mb/s , а в последните версии 100 Mb/s и 1 Gb/s. Използват се два метода на предаване: директно цифрово предаване (baseband); модулирано аналогово предаване (broadband). Вариантите на 10 Mbs Ethernet са 10 Base 5 - с дебел коаксиален кабел, 10 Base 2 - с тънък коаксиален кабел, 10 Base T - с усукани двойки проводници, 10 Base FL - с оптичен кабел, 100 Base T - Fast Ethernet и 1 Base 5 - STARLAN- 1 Mb/s. Цифрите и думите в означенията означават следното: Base - работа в основната лента в канала, 10 - скоростта на трансфер в Mb/s, 5 - максималната дължина на сегмент в стотици метри. За всеки стандарт си има комплект кабели и конектори.

Сега стандартът 10 Base 5 почти не се използва. За разлика от 10 Base 5, в стандарта 10 Base 2 приемо-предавателите се намират в самите мрежови адаптери. Използва се тънък коаксиален кабел тип RG-58. Стандартът 10 Base T използва кабел UTP (неекранирана усукана двойка проводници). Освен усукана двойка проводници се използват многопортови концентратори (английския термин за тях е hub). Структурата, която се прави с hub е звездовидна. Стандартът Fast Ethernet се означава с 100 Base - T, осигурява скорост на предаване на данните 100Mb/s, има звездна топология, метод за достъп CSMA/CD, използва UTP- проводник.

Стандартът Gigabit Ethernet осигурява скорост 1 Gb/s, има звездна топология, метод за достъп CSMA/CD и предава основно по оптичен кабел. Съвместим е със стандартите 10base-T и 100Base-T.

Концентраторите, използвани в Ethernet - мрежите са два вида-обикновени - просто осъществяват връзката между компютрите, и комутиращи (switching hubs).

На фиг.4.2 е показано каскадното свързване на няколко концентратора.



Фиг. 3.2

Вместо обикновен концентратор, може да бъде поставен комутиращ (съкратено "комутатор"). Комутаторът управлява всички съединения между компютрите и поради това няколко от тях могат да предават едновременно. Той се използва за намаляване на конфликтите в мрежи с интензивен трафик.

Всички кадри, които се предават по мрежата, имат адресна част. Комутиращите концентратори анализират адресната част на кадъра, където е адреса на получателя и кадъра "се превключва" към порта на получателя, докато другите портове на комутиращия концентратор остават свободни. В резултат в определен момент от време могат да се "включат" и обработят няколко пакета към различни портове. Това води до повишаване на общата пропускателна способност и вдигане скоростта на обмен в мрежата.

Производителността е количеството на потребителските данни, които мрежата може да предаде от един компютър към друг.

Стандарт IEEE 802.5 описва LAN с логическа топология тип "кръг". Разстоянията, които се покриват са по-големи от 802.3 и 802.4, тъй като когато сигналът преминава през възела се усилва. Стандартът използва само режим "директно предаване". Използват се кабелите: неекранирани усукани двойки (UTP) с конектори RJ – 45; екранирана усукана двойка (STR) с конектори RJ – 45 или с MIC на IBM; коаксиален кабел; влакнесто – оптичен кабел. Скоростта на предаване за стандарта е 16 Mb/s или 4 Mb/s. По кабел UTP се постига скорост до 4 Mb/s.

Крайните възли се свързват към кръга чрез специални концентриращи устройства MAU (Multistation Access Unit). Чрез съединителите MAU може да се изключва повреден краен възел, с което се запазва целостта на мрежата.

В една LAN може да има до 33 броя MAU. Разстоянията между MAU трябва да са: минимално 2, 5 м.; максимално – до 100 м за кабел STP и 45 м за кабел UTP.

В MAC – подслоя се използва протокола Token Ring. за работа в реално време.

В LAN с по-висока скорост (16 Mb/s) се използва методът на "предварително освобождаване на кадъра". Възелът – подател не чака кадърът да "направи кръг", а щом предаде кадъра в мрежата, веднага предава и маркера към следващия възел.

В мрежите по стандарта Token Ring също се използват обикновени и комутиращи концентратори. На практика кръговата топология става тип "звезда". Маркерът не е необходим повече т.к. всички станции могат да предават когато пожелаят, но формата на кадрите остава същия както при Token Ring.

Стандартът IEEE 802.12 (100 VG – Any LAN) е за скорост 100 Mb/s. Any LAN означава всякаква локална мрежа. Той поддържа кадри с формат CSMD/CD (Ethernet) и кадри с формат Token Ring, които са 90 – 95% от LAN технологиите на практика.

Технологията 100 VG – Any LAN се конкурира с Fast Ethernet, но фирмите-производители поддържат и двата стандарта. Стандартът IEEE 802.12, представен от своите протоколи, съответства на OSI – модела. Единствената разлика е в най-долния слой на стандарта, който е разделен на две: долен подслой PMD – зависещ от физическата среда; горен подслой PMI – независещ от физическата среда.

Много приложения от съвременното поколение софтуер (богати на трафика мултимедийни приложения от типа клиент/сървър, приложение за съвместна (групова) работа и др.), изискват при работа в мрежа по-високи скорости на обмен. Развитие на високоскоростните технологии за компютърни мрежи понастоящем е свързано с 4 различни високоскоростни стандарта на ЛКМ- FDDI (Fiber Distributed Date Interface) - компютърна мрежа с оптичен кабел и скорост на обмен 100 M bit/s, Fast Ethernet (100 Base-T) - еволюционно развитие на ЛКМ Ethernet (10 Base-T) със скорост на обмен 100

M bit/s, 100VG Any LAN - високоскоростен стандарт за комуникация със 100 M bit/s, обединяващ протоколите на Ethernet и Token-Ring, и ATM (Asynchronous Transfer Mode) - авангардна високоскоростна технология на КМ със скорост 155 M bit/s.

Топологията на мрежата е “двоен кръг”, като единият кръг е основен (първичен), а другият – резервен (вторичен). Посоката на предаване в двата кръга е различна.

Мрежовите възли са два типа: мрежови възли, свързани към двата кръга (DAS); мрежови възли, свързани към единия кръг (SAS). SAS – възлите се свързват към кръга чрез DAS- концентратори.

При нормален режим на работа данните преминават през всички мрежови възли и всички участъци на кабела на първия кръг. Затова той се нарича транзитен. Вторичният кръг в този режим не се ползва.

Освен за гръбнак (опорна мрежа) на свързани към нея по-нискоростни ЛКМ, FDDI се използва за локални мрежи, свързани с обработка на видео, и в системи за автоматизация на проектирането (CAD/CAM системи).

. 3.4 Безжични локални компютърни мрежи

Терминът **безжични връзки**, кореспондира с използването на мобилните телефони. Но има и други области, където удобството на безжичните комуникации ги е изтласкало на гребена на комуникационната вълна. Една от тях е свързана с технологичния пробив в областта на комуникациите - **безжичните мрежи**.

Използването на радиоканалите за връзка между устройствата, макар че не е ново като изобретение, едва напоследък еволюира значително, особено благодарение на широкото разпространение на Internet, локалните и WAN мрежи, свързващи много хора, позволяващи да си разменят глас, видео и данни помежду си, и то с високи скорости.

Комитетът по стандарти на организацията IEEE (Institute of Electrical and Electronics Engineers) разработи IEEE 802.11 - първият стандарт за безжични мрежи (WLAN) със скорости за предаване на данни от 1 и 2 Mbps. През септември 1999 година е ратифицирано разширението на стандарта, получило наименованието IEEE 802.11b . Основната разлика от предишния стандарт е повишена до 11 Mbps скорост на обмен на данни между безжичните устройства. Стандартът IEEE 802.11 работи в съответствие с двете долни нива на модела OSI - физическо и канално ниво. Всяко едно мрежово приложение, протокол или операционна система могат да работят при това положение в една безжична мрежа не по-лошо, отколкото това става в обикновена Ethernet мрежа. На физическо ниво са отделени общо три

метода за предаване на данни, единият от които е в инфрачервения диапазон, а другите два са радиочастотни, работещи в интервала между 2.4 GHz и 2.483 GHz.

Стандартът 802.11 фиксира два вида безжично мрежово оборудване - **клиент**, ролята на който обикновено се поема от компютър с инсталирана безжична мрежова интерфейсна платка (Network Interface Card, NIC), и **точка за достъп (Access point, AP)**, която служи за връзка между безжична и кабелна мрежа.

Клиентът е окомплектован с мрежова карта 802.11, която може да бъде с интерфейс ISA, PCI или PC Card, както и във вид на вградено решение. Точката за достъп обикновено е оборудвана с приемо-предавател, интерфейс към кабелна мрежа (802.3) и специализирано програмно осигуряване.

Режимът клиент/сървър предполага използването на поне една точка за достъп, представляваща специализирано устройство, която да е включена към кабелна Ethernet мрежа, и определен, често ограничен брой крайни безжични работни станции.

Повечето модели интерфейсни карти са предназначени за включване към шината PC Card/PCMCIA. За да могат те да бъдат монтирани в компютрите, които нямат такъв слот, производителите предлагат преходници към PCI от PCMCIA. За удобство на всички, на които нямат свободно място на PCI слотовете на desktop системите им, много от производителите произвеждат и външни устройства с интерфейс USB. Далечината на връзката е 100-500 м в зависимост от външните условия и от скоростта.

Логичното изискване към все по-големите обеми на трансферирана информация изисква нови промени в стандартите за безжични комуникации. Въведе се 802.11b, с три много основни "вътрешни" разлики: интерфейс: Card Bus, USB 2.0; скорост на предаване на данни: до 54 Mbps; работна честота: 5 GHz. Диапазонът от честоти, отделен за IEEE 802.11a, съвпада с европейския стандарт HIPERLAN (High Performance Local Area Network).

Макар че повечето от устройствата 802.11 и да разполагат с **вътрешни антени**, повечето от тях, особено точките за достъп, разполагат с възможност за включване на **външни антени**. Използването на външни антени, позволява да се увеличи обхвата на устройствата и да се стабилизира и усили приемания сигнал, като в крайна сметка разстоянието между безжичните устройства може да достигне десетки километри. Разбира се, с вътрешните антени е нереално да се очакват подобни разстояния, дори и при пряка видимост между безжичните устройства. Използваните антени могат да бъдат от няколко типа - от антени с тясни диаграми до кръгови (широконасочени) антени.

При всички положения, проблемът със защитата на данните при безжичните комуникационни устройства е открит - все още е сравнително

лесно да бъде уловен сигнала от ефира и той да бъде декодиран. Всички технологии за безжични комуникации използват един или друг вариант на кодиране на данните с цел тяхна защита. Мрежите, отговарящи на стандарта IEEE 802.11, използват функции за криптиране WEP за кодиране на информацията, като, в зависимост от класа на устройството криптирането може да бъде 64- или 128- битово.

При Bluetooth има три режима на защита, като най-защитеният Security mode 3 (link level enforced security) оперира със сеансови ключове (Bond), които се генерират в процеса на свързване на две устройства, и се използват в процеса на свързване, идентификация и предаване на данни между две устройства.

3. 5 Свързване на мрежи

Основен проблем, който възниква при междумрежовите комуникации, е свързването и съгласуването на хетерогенни (нееднородни) мрежи, т.е. мрежи, обхващащи различни по вид работни станции и сървъри, с различни операционни системи и приложни програми, и най-вече с различни комуникационни протоколите.

Съгласуването на две мрежи, използващи в горните слоеве на модела еднакви протоколи и различаващи се само по използваните в долните слоеве протоколи, се извършва с помоща на допълнителни устройства, поставени между тях. В зависимост от нивото, на което трябва да се извърши съгласуването, се различават следните устройства-посредници за междумрежови комуникации: повторители, концентратори, модеми, мостове, маршрутизатори, мост-маршрутизатори, комутатори.

Повторителите се използват за свързване на еднакви сегменти от ЛКМ. Тяхната функция е да регенерират и усилват входните сигнали. Те работят на физическо ниво в мрежите Token ring и Ethernet. Чрез тях се увеличава общата дължина на мрежата.

Концентраторите (Hubs) са много прости устройства, те работят на физическо ниво и пропускат целия трафик между секциите от ЛКМ които свързват и в двете посоки. Всички съобщения се придвижват до всички секции от ЛКМ, даже и да не е необходимо да отиват там. Концентраторите изпълняват същите функции като повторителите, но свързват повече от един сегмент на мрежата.

Мостовете се използват, когато в някои случаи една ЛКМ поиска достъп до данни или ресурси от друга локална мрежа. Мостът е по-сложно устройство от повторителя или концентратора, тъй като той трябва да прочете адреса на получателя от адресната част на постъпващия кадър и да реши към коя страна на моста трябва да изпрати този кадър. Мостът не променя кадрите, той се използва за свързване на еднакви локални мрежи, работи на канално ниво на модела OSI. Разделянето на ЛКМ чрез мостове има някои предимства. Така се увеличава общата надеждност на мрежата, тъй като повреда в един от сегментите няма да се отрази на останалите. Мрежата остава разделена на отделни сегменти, и производителността на мрежата се оптимизира чрез тези сегменти с по-нисък трафик. По този начин може да се отделят и различни типове трафик и да им се даде различно ниво на сигурност чрез различни физически линии. Производителността на моста се измерва чрез брой пакети в секунда

или кадри в секунда. Мостът трябва да прочете всеки пакет и филтрира тези, които трябва да се изпратят към другата страна на моста. Скоростта на филтриране варира от 7000 до 60 000 кадъра в секунда. Мостовите са сравнително прости устройства, те са лесно конфигуруеми и не оказват влияние върху софтуера на хост- компютъра. Те са подходящото решение за среда, където ЛКМ са еднакви.

Главните различия между повторителя и моста са следните. Повторителя се използва за удължаване на LAN чрез усилване на сигналите. Те се повтарят за всички сегменти, даже и ако не са предназначени за тях. Мостът чете адреса на получателя в пакета и го пропуска само ако е адресиран за друга LAN.

Маршрутизаторите са предназначени за свързване на две или повече локални мрежи, които имат еднакви или различни кабели и протоколи. Най-главната черта на маршрутизатора е неговата способност да избира най-добрия маршрут за предаването на данни между мрежите, ако тези маршрути са повече от един. Маршрутизаторите използват маршрутна таблица за откриване на най-добрия маршрут и извършват по-голяма по обем обработка на съобщенията, затова работят по-бавно от мостовите. Маршрутизатора трябва да се справи със значителните разлики в мрежите, които ще свърже. Адресните схеми могат да бъдат различни и даже може да имат различни по дължина адресни полета. Размерите на кадрите са различни, и кадрите от едната мрежа може да се наложи да бъдат разбити и да се съставят нови (това се нарича сегментация).

Маршрутизаторите са от 2 до 3 пъти по-скъпи от мостовите, те работят на мрежово ниво на OSI модела. Маршрутизаторите извършват по-голяма обработка. Най-главната разлика между моста и маршрутизатора, е че последния обработва само съобщения, адресирани до него и това, че може да се наложи той да пазцепи съобщението на по-малки кадри.

Ако възел от LAN тип Ethernet изпраща съобщения към възел в LAN тип Token Ring чрез опорна мрежа тип FDDI, се използват маршрутизатори (рутери) включени към мрежите. В този случай стандартите FDDI и ATM изискват преобразуване на пакетите от Ethernet и Token Ring мрежите, преди те да тръгнат по опорната мрежа (гръбнака).

Съществуват комбинации между мост и маршрутизатор- наричат се мост-маршрутизатори (B-routers). Те комбинират функциите на моста и маршрутизатора и работят на двете нива - канален и мрежов слой на модела OSI. Свързват еднакви или различни локални мрежи. Тяхното предимство се състои в това, че те са бързи като мостовите, когато ЛКМ са еднакви, и гъвкави когато трябва да свързват различни мрежи.

Шлюзовете са най-сложните устройства за свързване на мрежи, работят на по-горните нива на модела OSI- мрежово, а понякога и на приложно ниво. Те свързват ЛКМ към глобални компютърни мрежи и извършват преобразуване на протоколите и превръщане на скоростите, ако е необходимо. Шлюзовете премахват оригиналния протоколен комплект на изпращача и го препакетират. Те преобразуват мрежови протоколи и формати на данните. Освен това могат да бъдат използвани да свързват мрежи, които не са съвместими с OSI - модела. Шлюзовете използват всичките 7 - слоя на модела OSI. Те обаче могат да станат тясното място, когато трафика е много голям.

Опорна мрежа се нарича високоскоростна мрежа, която обикновено свързва в едно цяло всички мрежи, например на една корпорация, образувайки по този начин единна мрежа. Всички опорни мрежи трябва да имат способността да пренасят различни типове трафик, да пренасят различни протоколи за данни, да осигуряват връзки между локални мрежи, връзка с глобални мрежи, да работят надеждно и с такава скорост достатъчна да поддържа ефективно бизнес операциите и да бъде "прозрачна" за потребител.

3.6 Глобални компютърни мрежи

Глобалните компютърни мрежи (WAN) покриват големи географски разстояния (държави и континенти). Всяка от тях се състои от комутационни възли, свързани помежду си с комуникационни линии, топологията на мрежите е произволна. Глобалните мрежи използват някои от следните методи за комутация:

1. Комутация на канали - след заявка за връзка се установява временен канал между източника и получателя на съобщения, и той не се разпада, докато не завърши предаването на цялото съобщение. Този вид комутация се използва в съществуващите телефонни мрежи, и в новите цифрови ISDN - мрежи с интеграция на услугите.

2. Комутация на съобщения - съобщението се предава по различни участъци на мрежата с натрупване в паметта на междинните възли. При големи натоварвания се увеличава много времето за доставка на съобщенията заради образуването на опашки в претоварените междинни възли. Този метод се използва понякога за електронна поща.

3. Комутация на пакети - съобщенията (техния размер може да бъде десетки мегабайта), предавани от крайните устройства за данни (компютри), се разделят на малки части, наречени пакети, като във всеки от тях освен част от съобщението, се поставят адреси на предавателя и приемника. След това отделните пакети се предават от един комутационен възел към друг възел (възможно по различни маршрути), докато достигнат до получателя. Този метод има два режима - дейтаграмен и режим на виртуално съединение.

При **дейтаграмния** режим всеки пакет, наричан дейтаграма, се предава независимо от другите и по различен маршрут. Всеки възел (комутатор/маршрутизатор) избира на базата на предварително получена информация за състоянието на мрежата, към кой следващ възел да предаде пакета. За целта възлите имат така наречените маршрутни алгоритми и таблици.

При режима на **виртуално** съединение се изгражда логическа (виртуална) верига на свързване между крайните възли и след това се предават всички пакети на съобщението един след друг по реда на следването им. Този режим осигурява по-голяма скорост на предаване на съобщенията и предотвратява разместването на пакетите по сравнение с дейтаграмния, но тук не се прави контрол на задръстванията във възлите.

Комуникациите в глобалните мрежи използват аналогови и цифрови технологии за предаване и пакетно комутиране. Аналоговите линии са телефонните линии, използвани от обществената комутируема телефонна мрежа. За ускоряване на предаването се използват арендовани аналогови линии, но тяхното използване е по-скъпо.

Потребители, които генерират голям трафик за глобалните мрежи, е по-добре да използват линии за цифрови данни, при които скоростта е много по-голяма и предаването е почти безпогрешно. В САЩ и Европа се използват цифровите арендовани линии T1 и E1, предаващи по медни проводници речева информация, данни и видео информация със скорост 1.544 Mb/s. Арендваната линия T3 позволява скорост 44.736 Mb/s, а синхронната оптична мрежа SONET - скорост от 51.84 Mb/s до над 1GB/s.

Стандартът **X.25** е най-използваният за глобални мрежи с комутация на пакети. Той осигурява добро маршрутизиране във всеки един момент от време и скорост на предаване 64 Kb/s по аналогови линии. Между компютрите и комуникационната подмрежа се включват устройства, наречени PAD. Те получават по асинхронен път

данни от компютъра и ги превръщат в пакети, които се предават по мрежата. **X. 25** е приложен за WAN с комутация на пакети за скорости до 2 Mb/s . Стандартът определя интерфейса между крайния възел и глобалната подмрежа.

Модерните стандарти за глобални мрежи са по-надеждни. Един от тях е **Frame Relay**. Той определя виртуална наета линия, което е добре за компютърния трафик. Осигурява скорост на предаване на данните, речева и видео информация от 2 Mb/s до 34 - 45 Mb/s. Този стандарт се използва в глобални мрежи за предаване на анизохронни данни (неречева информация, компютърни данни), за свързване на локални мрежи по между им, и като средство за достъп до АТМ- мрежи.. Стандартът FR е от ново поколение за достъп до глобална мрежа с комутация на пакети. В началото той е част от ISDN, но по-късно намира приложение в частните мрежи. FR – стандартът може да се разглежда като виртуална наета линия. Абонатите наемат през мрежата постоянно виртуално съединение между два и повече крайни възела. Между възлите информацията се предава във вид на Frame Relay кадри. FR е по-добър от X. 25 по отношение на скорост на предаване: до 34 Mb/s в Европа; до 45 Mb/s в САЩ и Канада. Времето за закъснението на кадрите в мрежата е по-малко в сравнение с X. 25.

Стандартът АТМ е създаден от телефонната индустрия. Известен е като Cell Relay (Комутация на клетки). Използва се за предаване на данни блокове с фиксирана дължина 53 байта – наречени клетки. Това позволява по-бързото им предаване по мрежата и намалява закъснението поради по-ефективното комутиране, по сравнение с пакетите с големи размери. АТМ е стандарт не само за глобални мрежи, но се използва и в локалните мрежи, за високоскоростно предаване на реч, данни, видеоизображение, звук с качество на CD, графични изображения и др. АТМ- мрежите са интелигентни, те разпознават различните типове данни и ги обработват приоритетно. Скоростта на предаване на данните се движи от 155 Mb/s до 622 Mb/s. Стандартът АТМ е сроден със стандартите X. 25 и FR . Той използва комутация на клетките и мултиплексиране на няколко логически съединения по един физически интерфейс. Използват се предимствата на новите цифрови линии, като се осигурява по-бърза комутация на клетките

Стандартът **ISDN** (Integrated Services Digital Network) описва цифрова мрежа с интеграция на услугите. Той осигурява пренасяне в единен цифров вид на всякакъв вид информация- компютърни данни, глас, изображения, видео, факсимилна информация, музика. ISDN се развива на базата на цифровата телефонна мрежа. По съществуващите аналогови телефонни канали скоростта на предаване е 64 Kb/s, но тези линии се заменят с цифрови с голяма скорост.

Най-голямата **глобална мрежа Internet** обединява стотици милиони компютри от всички континент. Общото за всички компютърни мрежи, обединени в Internet е комуникационния протокол TCP/IP. Той е програмна реализация на съвкупност от правила и условия, която реализира обмена на данни между компютрите чрез пакетна комутация. Благодарение на този протокол компютрите в състава на Internet могат да общуват помежду си независимо от различията в техния клас, производител, операционна система и др. Протоколът TCP/IP се състои от два основни компонента - Internet Protocol (IP) и Transport Control Protocol (TCP). IP отговаря за адресирането, разделянето на съобщението на фрагменти, съответстващи на големината на пакетите, предавани в различни мрежи. TCP отговаря за сигурността при предаването на

данните. При създаване на Internet корпоративна мрежа се използва протоколния стек TCP/IP. Тези мрежи използват транспортните услуги на Internet и хипертекстовите технологии WWW. Използваните съвременни операционни системи Windows, Unix и Linux поддържат стандартно протоколния стек TCP/IP.

Комуникационният модел на протоколния стек TCP/IP е създаден също като отворен стандарт, но има различна философия с OSI - модела. В TCP/IP модела са обособени **четири слоя** и разделението между тях не е стриктно както в OSI – модела. Маршрутизацията в TCP/IP – модела също е функция на мрежовия слой, но маршрутизиращите протоколи RIP и OSPF използват протокола UDP, който е от транспортния слой.

Каналният слой (Host – to – network Layer) на модела TCP/IP съответства на каналния и мрежов слоеве на OSI – модела и поддържа всички стандарти за локалните и глобалните компютърни мрежи. Използва се за осъществяване на достъп до компютър, който е част от локална или глобална компютърна мрежа и е свързан към Internet. При осъществяване на комутиран достъп до Internet посредством модем се използват протоколите SLIP и PPP.

Мрежовият слой (Internet Layer) е слой на междумрежовото взаимодействие. Чрез него крайните възли вкарват пакети с данни в произволна мрежа и ги доставят до предназначения краен възел независимо един от друг. Горните слоеве на модела проверяват и възстановяват правилният ред на следване на пакетите. Мрежовият слой съставя маршрута за движение на пакетите от данни от един хост до друг хост, преминавайки през една или няколко свързани мрежи посредством рутери. Сложно разглежда пакетите като IP – дейтаграми. Използва се IP – протокол, който работи много добре в компютърни мрежи със сложна топология. Протоколът IP е дейтаграмен протокол и не дава пълна гаранция за доставка на пакетите с данни до получателя.

Протоколът IP е реализиран във всички крайни (Hosts) и междинни (Routers) на компютърната мрежа.

Транспортният слой (Transport Layer) на модела TCP/IP скрива детайлите по реализацията на връзката от приложните процеси и те нямат задължение към функции, като Скорост за обмен, Кодови таблици, Разделяне на съобщенията на пакети. Този слой използва протоколите TCP и UDP, които осигуряват предаване на пакетите с данни от край до край, посредством изграждане на виртуални съединения. Използва се метода на хлъзгачия се прозорец с положителна квитанция, като предаването е по дуплексен канал. Протоколът TCP в хоста подател приема съобщението от приложния слой, фрагментира го на сегменти и го изпраща по интермрежата посредством протокола IP. Протоколът TCP в хоста получател събира сегментите и възстановява съобщението.

Основни функции на протокола TCP са възстановяване на изгубен сегмент, възстановяване на реда за следване на сегментите, премахване на сегменти дубликати, управление на потока от данни.

Реализираните протоколи в хостовете и IP рутерите поддържат връзка между приложните процеси на крайните устройства. Комуникацията се осъществява чрез дуплексен канал по логически съединения между портове.

Комуникацията между два приложни процеса от различни крайни устройства започва като приложният процес на хоста – подател извиква протокола TCP и му предава данни. Най-напред протоколът TCP изгражда логическо съединение между приложните процеси, които ще си обменят данни. Протоколът използва номера на портове за да укаже кой от стартиралите процеси в хоста предава данните. Следва разделяне на получените данни на TCP – сегменти с подходящи размери за мрежата, към която е свързан хоста подател. В заглавната част на TCP – сегментите се помещава служебна информация за метода за контролиране на грешките и за реда на следване на сегментите.

Предаването започва, след като протоколът TCP извиква протокола IP, който капсулира сегментите и ги превръща в IP – дейтаграми. IP – дейтаграмите преминават през серия от IP – рутери, за да достигнат до IP – модула на хоста получател.

Хостът подател вмъква своя IP – адрес и адреса на хоста- олучател в заглавната част на сегментите IP – адресът се състои от две части: Идентификатор на мрежата (Net ID); Идентификатор на хоста (Host ID).

Малките локални мрежи имат само един IP – рутер и всички дейтаграми за другите подмрежи се предават през него, след което преминават през серия от IP – рутери до пристигането си в IP – рутера на мрежата, в който се намира хоста получател. Когато дейтаграмите преминават през междинните рутери, кадърът на предходната мрежа се премахва, анализира се заглавната част и се определя следващия път по маршрута. Дейтаграмата задължително се опакова с кадър, съответстващ на следващата подмрежа и т.н..

Всеки рутер има възможност да раздели дейтаграмата на по-малки части- “фрагменти” в зависимост от изискванията на следващата подмрежа. Това се налага, за да може полето <данни> да се вмести в кадъра на каналния слой. Не е допустимо в междинните рутери да се извършва дефрагментиране. Този процес се извършва единствено в хоста получател. IP – модулет на хоста-получател дефрагментира TCP – сегментите и ги предава на TCP – модула. Протоколът TCP поставя данните в буфери, след което известява приложния процес за пристигането им. Всички IP – дейтаграми се контролират от междинните възли на интермрежата по време на предстоя им в тях. Във всяка от тях има поле, в което е указана продължителността на живот, който има дейтаграмата в интермрежата. Стойността на това поле намалява с единица, преминавайки през междинен рутер и при изтичане на всяка секунда. Тогава, когато плето <време на живот> се изчерпи, дейтаграмата се унищожава и на хоста подател се изпраща съобщение. Коректното предаване на данните се проверява от шестнадесет битова контролна сума, която обхваща заглавната част и полето данни на дейтаграмата. Контролната сума се проверява от всеки рутер, през който преминава дейтаграмата и в хоста получател. При откриване на грешка дейтаграмата се бракува и се изпраща съобщение на хоста-подател.

Независимо от общите черти между моделите TCP/IP и OSI съществуват и различия, заключаващи се в следното:

- Понятията услуги, интерфейс и протокол в модела OSI са рязко разграничени за разлика от модела TCP/IP;
- Моделът OSI е проектиран и след това са създадени протоколите му. Практическото приложение на протоколите показва, че те не съответстват напълно на модела и се наложило въвеждане на ДОПЪЛНИТЕЛНИ ПОДСЛОЕВЕ. Моделът TCP/IP възниква след създаването на протоколите, като тяхно описание;
- За разлика от модела OSI моделът TCP/IP може да се използва само за описание на TCP/IP мрежи;
- Моделът OSI поддържа комуникации с и без установяване на логическо съединение в мрежовия слой и само с установяване на логическо съединение в транспортния слой. Моделът TCP/IP поддържа комуникации само с установяване на логическо съединение в мрежовия слой а в транспортния слой и двата вида комуникации;
- Моделът TCP/IP се състои от четири слоя, а моделът OSI от седем;
- За разлика от модела OSI, моделът TCP/IP не е строго иерархичен. Някои протоколи в TCP/IP от горните нива се ползват директно от долните –

протоколът PING от приложния слой използва мрежовия протокол ICMP без посредничество на междинен протокол;

- Протоколът TCP/IP се развива основно от потребителите, докато моделът OSI се развива от доставчиците, тъй като той е официален международен стандарт.

Освен сървъри и хостове, в Internet има специализирани компютри, които управляват предаването на данни (трафика) в мрежата - маршрутизатори, и устройства за свързване на мрежи една с друга - мостове и шлюзове. Ако информацията се движи между две мрежи, използващи един и същ протокол, маршрутизатора е устройство, което ги свързва и маршрутизира пакетите. Ако мрежите са изградени на основата на различни протоколи, тогава се използва шлюз. Той преобразува информацията от пакетите от единия протокол, в пакети на другия протокол.

В модела TCP/IP се използват четири вида адресации в съответствие с броя на слоевете му. Тези адресации са следните: Адресация по локален адрес; Адресация по IP адрес; Адресация по порт; Адресация по DNS символно поле.

Адресация по локален адрес се използва от каналния слой и се определя от технологията на функциониране на мрежата, в която се намира хоста. За локалните мрежи това е MAC – адреса на мрежовия адаптер на хоста или IP – рутера. MAC – адресът на мрежовия адаптер се назначава от фирмата производител на мрежови адаптери и е уникален. Старшите три байта на адреса са идентификатор на фирмата производител. Младшите три байта се дават от фирмата производител и са уникални.

Пример за MAC адрес на мрежов адаптер: 10-B1-06-2C-E1-11.

Локалните адреси на глобалните компютърни мрежи се назначават от администраторите на мрежите.

Адресация по IP – адрес се използва от мрежовия слой. Такъв адрес е уникален и се отнася за краен възел или порт на рутер. IP – адресът се назначава от администратор при конфигуриране на компютрите и рутерите. IP – адресът се състои от две части: Адрес на мрежата (Net ID); Адрес на хоста (Host ID).

Адресът на мрежата може да се избере произволно от администратор или да се даде от специалните адреси на Internet центровете които са:

- INTERNIC за Северна Америка;
- RIPE – NCC за Европа;
- AP – NIC за Азия.

Локалния адрес на хоста и IP – адресът са независими. Разделението между адреса на мрежата и адреса на хоста е гъвкаво, като границата между тях е произволна и се определя с помощта на “подмрежова маска” (subnetwork mask). Допуска се един краен възел да влиза в състава на няколко IP – мрежи ако има съответните IP – адреси. Мрежовото съединение се определя от IP – адреса и не определя компютъра или рутера. IP – адресите са четири байтови и се записват с четири десетични числа, разделени с точки. В IP – протокола са предвидени някои специални адреси като:

- Когато целият адрес е съставен само от двоични нули, се указва хоста, който е генерирал пакета;
- Ако само частта Net ID от IP – адреса на хоста – получател се състои от “0₂”, той се намира в мрежата на хоста – подател;
- Ако целият IP – адрес на хоста – получател е от “ 1₂” то IP – дейтаграмата е предназначена за всички хостове, намиращи се в подмрежата на хоста – подател (предаване до всички);

- Ако само Host ID от IP – адреса на хоста – получател се състои от “ 1_2 ”, това означава, че IP – дейтаграмата е предназначена за всички хостове на подмрежата с посочения Net ID;
- Ако само Host ID от IP – адреса на хоста- получател се състои от “ 0_2 ”, то IP – адресът указва подмрежата със зададения Net ID;
- IP – адресът 127.0.0.1 се използва за тестване на протоколния стек TCP/IP, реализиран в дадения хост.

Четвъртата версия на IP – протокола (IP.v.4) определя пет класа IP – адреси (А, В, С, D, Е).

| | | | | | | | | | |
|---|---|--------|--------|--------|---------------------------------|--------------------------------|---------|---------|--|
| | 0 | 7 | 8 | 15 | 16 | 23 | 24 | 31 | |
| A | 0 | Net ID | | | | Host ID | | | |
| B | 1 | 0 | Net ID | | | | Host ID | | |
| C | 1 | 1 | 0 | Net ID | | | | Host ID | |
| D | 1 | 1 | 1 | 0 | Групов адрес (multicast) | | | | |
| E | 1 | 1 | 1 | 1 | 0 | Запазен за бъдещи нужди | | | |

Фиг.4.4 Структура на IP – адреси на протокол IP v. 4.

Първият байт на IP – адрес клас ”А” започва с “ 0_2 ” и адресите от този клас започват от 1 до 126. Адресът на Net ID заема един байт, а останалите три байта са адрес на Host (Host ID). Клас А адресите се използват за големи мрежи с много крайни възли. Десетичната нула (“ 0_{10} ”) е първото число на IP адресите, а адресът започващ със 127 се използва само за специални цели. IP адресът 127.0.0.1 се използва за тестване работата на TCP/IP – стека в един възел, без да се изпращат пакети в мрежата.

Първият байт на IP – адрес клас ”В” започва с $(10)_2$. IP –адресите започват с $(128)_{10}$ и завършват с $(191)_{10}$ включително. Този клас адреси се ползва за мрежи със среден размер, на университети и по - големи компании.

Първият байт на IP – адреси клас ”С” започва с комбинацията $(110)_2$. IP – адресите започват от число $(192)_{10}$ до $(223)_{10}$. Използва се за локални мрежи. Net ID заема 3 байта, а Host ID – 1 байт. В такава мрежа може да има до 254 хоста.

Първият байт на IP – адреси клас ”D” започва с 1110. IP – адресите са в диапазона от $(224)_{10} \div (239)_{10}$ включително. Използват се за изпращане на съобщения до група хостове, на които е присвоен уникалният адрес. Този адрес не се дели на Net ID и Host ID.

Първият байт на IP - адреси клас ”Е” започва с двоичната комбинация $(11110)_2$. IP – адресите са в диапазона $(240)_{10} \div (254)_{10}$ включително. Този клас адреси са запазени за бъдещи приложения.

В IP – адресацията се използват “подмрежови маски”, когато адресите Net ID не достигат за да се структурират.

Маската е четири байтово число, двоичният запис на което съдържа “1”₂ в битовете, представляващи адреса на мрежата (Net ID) и “0”₂ в Host ID.

- за клас А маската е 255.0.0.0;
- за клас В маската е 255.255.0.0;
- за клас С маската е 255.255.255.0.

Портът е вид адреси и се използват в транспортния слой. Портовете се използват от протоколите TCP и UDP за връзка с приложните процеси. Портът е 16 битово число. Портовете имат конкретно предназначение в компютърните комуникации и са обвързани със съответен протокол. Порт 21 е стандартно резервиран за протокола FTP а порт 23 за протокола TELNET. Понеизвестните протоколи използват свободно избрани незаети номера за портове. Портът и IP - адресът съвместно образуват сокет (socket). IP адресът 193. 68. 180. 5. и порт 21 идентифицират TCP – съединение. Един сокет може да участва в няколко съединения едновременно.

Приложният слой използва DNS (Domain Name System) – имена. Това са имена на хостове и са въведени за удобство на потребителите. Те се помнят по-добре в сравнение с цифрите на IP – адресите. DNS – имената се използват от мрежовите администратори. Състоят се от части, разделени с точки, като старшият домейн е отдясно. Домейнът в Internet е логическо обединение на хостове от една или различни мрежи. Използват се поддомейни за по-лесно администриране.

Синтаксиса на DNS – имената е:

Поддомейн N. Поддомейн N – 1 ... Поддомейн 1. Домейн.

Пример за DNS – име е WWW. fmi. shu – bg. net

- fmi - поддомейн на shu-bg и домейн на fmi;
- shu – bg – поддомейн на net домейн на fmi.

Имената са част от разпределена база от данни, наречена DNS. Поддържа се йерархична система от имена за идентификация на хостовете и ресурсите в Internet. Основната роля на DNS е търсене на IP – адреси по съответните им DNS – имена. За целта се използва протокола DNS на приложния слой. Протоколът DNS е асиметричен. В него са определени DNS – сървъри и DNS – клиенти.

DNS – сървърите съхраняват части от базата данни за съответствие на DNS – имена и IP – адреси. Тази база от данни е разпределена по административните домейни на Internet. базата данни на всеки DNS – сървър се нарича още master – file или zone – file.

По правило съдържанието на базата данни в DNS – сървър е станична и се актуализира три пъти седмично от съответния за даден континентален информационен център (RIPENIC- за Европа). Разпраща се актуализирана информация до всички DNS – сървъри чрез Internet.

DNS – клиентът е програма, която се стартира на компютър, който трябва да разбере даден IP – адрес. DNS – клиентите знаят IP – адреса на DNS – сървъра от своя домейн. DNS – сървърите и DNS клиентите постоянно кешират информацията, предоставяна по заявките. По този начин, при следващите търсения времето за достигане до тази информация се намалява.

DNS – сървърите са съединени помежду си в съответствие с йерархията на домейните, които обслужват. Коренът на DNS - йерархията е в континенталния център и той няма име. Домейните от първо ниво са обикновено държавните домейни или на организационна основа.

Всеки DNS - домейн се администрира от отделна организация, която я разделя на поддомейни и предоставя администрирането на други организации.

За България това е фирма “Цифрови системи” – Варна за домейна bg. За цяла Европа общият регистър се поддържа от RIPE – Амстердам.

Всеки домейн и поддомейн имат уникално име – до 63 символа. Пълното DNS – име може да е до 255 символа.

Поради проявяващия се вече недостиг на IP адреси по сегашния протокол IPv4, има ново съглашение- IP (v6) (или IPng - IP Next Generation), в което адресите стават 128 битови и това ще осигури 340 282 366 920 938 463 463 374 607 431 768 211 456 адреса. Този протокол осигурява съвместимост с протокола IPv4.

3.7 Достъп до глобалната мрежа Интернет от стационарни и мобилни компютри

Обикновено за достъп до ресурсите на Интернет потребителя ползва стационарен персонален компютър, включен към дадена локална мрежа (в Интернет клуб, учебна зала и др.). Там се предполага , че компютъра има необходимите настройки и на потребителя остава само да стартира съответната клиентска програма . Ако обаче връзката не се установи , след проверката за правилно кабелно свързване може да стане ясно, че е необходима нова настройка, и потребителят трябва да я извърши. Това не е сложна процедура.

Под управлението на ОС Windows XP се избира последователно **SETTINGS – CONTROL PANEL – NETWORK AND INTERNET CONNECTIONS- SET UP OR CHANGE YOUR INTERNET CONNECTIONS- INTERNET PROPERTIES- CONNECTIONS- LAN SETTINGS:**

Набира се в **Proxy server**, например 10.10.0.1 и номер на порт- 3128 (това не се отнася при модемна връзка и за VPN връзка).

Тази настройка може да се направи и чрез браузъра **MS IE** и последователността на избор **TOOLS- INTERNET OPTIONS- CONNECTIONS- LAN SETTINGS.**

В браузъра трябва да се отвори съответния прозорец и да се провери IP адреса на **шлюза**, чрез който локалната мрежа се свързва с Интернет-доставчика, IP адреса и стойността на маски, подмаски и др.Те трябва да се получат от съответния системен администратор. Трябва да се въведе и вида на протокола-TCP/IP.

Все по- широкото навлизане в бита на безжичните Интернет комуникации води до покриване на райони на организации, летища, гари и даже на цели селища с т.н. зони за безжична интернет връзка. У нас Министерството на транспорта и съобщенията реализира такива зони с безжичен и безплатен интернет достъп, наречени **i- zone** . В тези зони чрез обикновен компютър или лаптоп може да се получава свободен достъп до мрежата. Студентите са първите, които използват тези зони. В тези зони може да се използва не само компютър, но и PDA (Personal Digital Assistant).

Повечето от новите преносими устройства вече се произвеждат с вградена **WiFi** карта. Ако компютъра няма такава , трябва да се използва **WiFi** адаптер за безжичен достъп през USB порт или PCMCIA карта.

Практически съвети:

1. Проверява се дали компютъра (устройството) има хардуер за безжичен (**WiFi**) достъп.
2. Проверява се дали този хардуер е активен.
3. При включване на компютъра (PDA и др.) чрез съответната програма се виждат какви безжични мрежи са достъпни. Избира се една от тези мрежи.
4. Отваря се браузър и се зарежда произволен сайт. Следват се инструкциите на екрана за включване в Интернет.

3.8 Сигурност на мрежите

Сигурността е свойство на една система да противостои на външни или вътрешни дестабилизиращи фактори, които могат да доведат до нейното нежелателно състояние или поведение. Това важи и за сигурността на информационните системи.

Компютърната и мрежова сигурност е свойство на компютърните системи и мрежи да противодействат на опитите за несанкциониран достъп до обработваната и съхраняваната информация, водещи до деструктивни действия и получаване на лъжлива информация.

Има три главни аспекта на сигурността на компютърните системи и мрежи - уязвими места, заплахи и противодействие.

Под уязвимо място се разбира всяка точка на компютърните и комуникационните системи и на системата за тяхната защита, където те са слабо защитени срещу заплахи и атаки, свързани с тяхната сигурност. Уязвимите места в една система зависят от конкретната ѝ реализация.

Всеки път, когато един компютър от компютърната мрежа поиска услуга от друга, тя посочва конкретен получател и метод на транспортиране. Получателят е посочен чрез адреса на получателя, а методът на транспортиране - чрез транспортния протокол. Освен това изпращачът указва приложението (програмата услуга) на машината получател, към която се обръща. Това става чрез системата за портове. Всяка услуга в интернет има присвоен уникален адрес, наречен порт. Портовете могат да бъдат отворени или затворени, през тях информацията може да влиза или излиза. Ако всички портове на компютъра са затворени, никой не може да проникне в него през мрежата, но и потребителят няма да може да използва услугите. В интернет сървър има хиляди портове, макар че повече от тях не са активни. Има програми, които разпознават кои портове са отворени и кои са затворени. Те се наричат скенери за портове. За защита е добре да се ползва програма като Zone Alarm от <http://www.zonelabs.com/>, която ще затвори всички портове на компютъра, които не се ползват от одобрената от потребителя програма.

Първата линия на отбраната срещу нарушителите е **системата от пароли**. Всички многопотребителски системи изискват от потребителите да се идентифицират с потребителско име (ID) и парола. Тази комбинация от реквизити на потребителя се нарича общо "акаунт". Една от заплахите за обектите на информационната система е неоторизираният достъп - когато неупълномощен потребител действа като законен. Това става, когато се използват общи, разгадани и прихванати пароли.

В много от протоколите данните се предават в открит, нешифрован вид. Анализ на мрежовия трафик позволява прехващане на данните, предавани чрез протоколите FTP и телнет (идентификатори и пароли на потребители), HTTP (предаване на хипертекст между уебсървъри и браузъри, в това число и въвежданата във форми информация от потребителите), SMTP, POP3, IMAP, NNTP (електронна поща и конференции) и IRC (online разговори, чат). Така могат да бъдат прехванати пароли за достъп до системите за електронна поща с уебинтерфейс, номера на кредитни карти при работа със системи за електронна търговия и различна информация от личен характер, разгласяването на която е нежелателно.

DoS (отказ на услуга) е вид атака, която причинява загуба на услуга или невъзможност на мрежата да функционира. Въпреки че чрез нея не се получава неоторизиран достъп до информацията на жертвата, тя си остава една от най-разпространените мрежови атаки. DoS атаките имат много форми и могат да бъдат насочени срещу всеки потребител в интернет. По принцип те работят по един от следните начини:

консумират пропускателната лента;
изчерпват ресурсите;
сриват системи и приложения.

С бурното развитие на WWW силно се увеличи броят на атаките през мрежата. Като цяло типовете атаки през WWW могат да се разделят на две големи групи:

Атака срещу клиент.

Атака срещу сървър.

Към настоящия момент най-широко разпространение имат т.н. **уебпробиви**. Този вид атаки се извършват от автоматично изпълними програми, чието предназначение е кражба или разрушаване на данните в компютъра. Те могат да бъдат инсталирани в клиентския компютър при сърфиране в интернет и изтегляне на необходими файлове от чужди сайтове или най-често при ICQ или IRC сесии. Този тип програми могат да бъдат Java аплети, ActivX обекти, Java скриптове, Visual Basic скриптове или практически всеки нов програмен език, предназначен за проектиране на уебстраници. Използват се често за компроментиране на конкретни мрежи или компании. Точна класификация на този тип атаки не може да бъде направена, тъй като с развитието на компютърните технологии се създават все повече нови начини за атака.

Спам е термин, влязъл в употреба през 90-те години. С него се означават съобщения, които наводняват мрежата и които хората никога не биха искали да получат, ако предварително ги бяха попитали. В тесен смисъл на думата спам е електронно съобщение с търговски или рекламен характер, което съдържа информация за стоки и услуги или приканва адресата да посети уебстраница с търговско съдържание.

Политика на сигурност включва множество правила, които определят как една организация управлява, защитава и разпределя класифицирана и друга важна информация. Това е рамката, в която една система осигурява защитата. Комбинацията от всички защитни механизми (или целостта им) в една компютърна система, включваща хардуер, фирмуер и софтуер, реализира политиката на сигурност.

Политиката за сигурност за компютърните мрежи на една организация трябва да обхваща мерки най-малко в следните области:

- физическа защита на отделните работни места и мрежата;
- организационни мерки, определяне на правата и задълженията на персонала;
- мерки за документална сигурност;
- политика на достъп до ресурсите (нива на достъп, пароли);
- използване на системи за кодиране на информацията;
- мониторинг и санкции за нарушенията на политиката на сигурност и др.

Средствата за реализиране на тези мерки са:

- административни - обхващат подбора на персонал, всички нареждания, правила и практики за сигурността;
- организационни - осигуряват защитата при проектиране и разработка на системата, както и функциите, правата и задълженията на персонала в организацията;
- технически и програмни - осигуряват защита на данните при глобалните комуникации, сървърите, локалните мрежи и работните станции, тук влизат и средствата за документална сигурност, мониторинг и физическа защита на елементите на системата.

За защита на данните в разпределените компютърни системи са разработени множество средства:

- защитните мрежови криптопротоколи;
- програмно-апаратни шифратори на мрежовия трафик;

- методиката "защитна стена" (Firewall);
- програмни средства за откриване на атаки (IDS - Intrusion Detection Systems или ICE - Intrusion Countermeasures Electronics);
- програмни средства за анализ на защитеността;

Всеки включен в интернет компютър предполага наличието на рискове за несанкциониран достъп до неговите ресурси. За защита срещу такъв достъп може да се използват възможностите на **защитни стени (Firewalls)**. Целта на защитната стена е да защити мрежата от външни атаки. Задачата на защитните стени се формулира по следния начин. Ако има две информационни системи или две множества от информационни системи, защитната стена е средство за разграничаване на достъпа на потребители от едното множество системи до информацията, съхраняваща се на сървърите на другото множество. Защитната стена изпълнява своите функции, като контролира целия информационен поток между тези две множества от информационни системи, изпълнявайки ролята на "информационна мембрана". В този смисъл защитната стена може да се представи като набор от филтри, анализиращи информацията, преминаваща през тях, и на основата на вложени в тях алгоритми, вземащи решение: да пропуснат тази информация или да откажат нейното пренасяне. Освен това тя може да регистрира събития, свързани с процесите на разграничаване на достъпа. В частност може да фиксира всички "незаконни" опити за достъп до информацията и допълнително да сигнализира за ситуации, изискващи незабавна реакция, т.е. да вдига тревога.

ТЕСТ

1. Аббревиатурата за локална компютърна мрежа е:
 а) WAN б) LAN в) WWW г) MAN
2. Аббревиатурата за глобална компютърна мрежа е:
 а) Internet б) WAN в) LAN г) PPP
3. Броят на нивата в отворения модел за комуникации е:
 а) 4 б) 6 в) 7 г) 8
4. Аббревиатурата PDU означава:
 а) компютър б) мрежа в) протоколна единица г) възел
5. Стандарт IEEE 802.3 се отнася за:
 а) ISO б) WAN в) LAN г) ITU
6. Топологията на локална компютърна мрежа може да бъде:
 а) куб б) звезда в) квадрат г) дъга
7. Стандарт за изграждане на компютърна радиомрежа е:
 а) 802.3 б) 802.5 в) 802.11 г) 802.12
8. Предаването на данни на големи разстояния става с:
 а) комутатор б) рутер в) шлюз г) модем
9. Кой от стандартите се използва за изграждане на глобални мрежи:

- A) Eternet B) X.25 C) TokenRing D) FDDI

10. Има ли няколко сървъра на един компютър?

- а) не;
б) да;
в) зависи от компютъра;
г) зависи от мрежата.

Глава4. WEB-ТЕХНОЛОГИИ

4.1 Web-сайтове и Web-браузъри

Internet дава възможност за получаване на теоретически безкрайна информация от най-различни области от цял свят - новини, справки, електронни публикации (статии, доклади), получаване на разнообразен (в това число и безплатен) софтуер, комуникации с абонати от мрежата, видеоконференции, слущане на радио, провеждане на телефонни разговори и др. Internet вече се използва и за електронна търговия.

Определено **услугата WWW (Web)** е най-атрактивната и най-добра относно разнообразието на информационни материали. WWW е постоянно променяща се колекция от огромен масив документи, всеки от които се намира някъде в Internet. Общото между тези документи е, че те са написани на езика HTML (Hyper Text Markup Language), за това се наричат хипертекстови. Това ги прави независими от особеностите на конкретната клиентска програма.

Хипертекстът е вид текст, който позволява вграждане на връзки към други документи. WWW е система за представяне на хипермедийна информация от влизаци в състава на Internet компютри, наричани Web-сървъри. Под хипермедийна информация се разбира мултимедийна информация с хипервръзки - съвкупност от текст, графични изображения, видеоклипове, звукозапис и анимация. В това отношение това е най-добрата услуга в Internet.

Началото на WWW е през 1989г. в института CERN от Тим Бърнърс Лий, който предлага идеята за разпределената хипермедия. Той и досега ръководи внедряването на различни стандарти за WWW чрез консорциума W3C (Word Wide Web Concorcium - <http://www.W3.org>).

Отделните компютри в Internet трябва да имат мрежов адрес и мрежово име, които да са уникални в рамките на цялата мрежа, за да могат еднозначно да се идентифицират източниците и получателите на информация от WWW. Мрежовият адрес се нарича още IP-адрес. Пряк достъп до Internet имат компютрите със собствен IP-адрес и които са независими възли в мрежата. IP-адресът се състои от 4 групи от числа, които се разделят с точка и имат различно предназначение, например **194.141.63.129**. Мрежовото име е символично означение на мрежовия адрес за по-лесно запомняне и идентифициране на устройствата от хората. Така например **www.uni-shoumen.acad.bg** се идентифицира с дадения по-горе IP-адрес. Мрежовите имена са буквени и се основават на **DNS** (Domain Name System) - йерархична система от имена на области (домейни). букви.

За получаване на информация от Internet трябва да се знае адресът на източника й. Този адрес се нарича **URL** (унифициран локатор на ресурси). Той има до 4 части-име на протокола за свързка със съответния сървър, например http или https за www, а за другите услуги - ftp, telnet и т.н.; адрес на сървъра в Internet; директория в сървъра,

съдържаща документа; име на файла, съхраняващ документа(ако това е Web-страница, то е с разширение html).

Web-сайт се нарича съвкупност от хипертекстови документи- Web-страници , обединени с един URL. Той се открива от клиента чрез този URL, представляващ електронния адрес на Web-сайта. Всяка от страниците е отделен файл с разширение **html**. Първата страница от даден Web-сайт се нарича "Home page". Тя го идентифицира и предоставя информация за този сайт (например таблица със съдържание) , съдържа препратки и хипервръзки до друга информация. **Web-мастер** се нарича лицето, създало Web-сайта.

Хипервръзката е препратка към други хипертекстови документи. Тя се изобразява чрез икона или подчертан текст с друг цвят на екрана. Щракването с мишката върху дадена хипервръзка предизвиква автоматична свързване с друга страница. Хипервръзките могат да бъдат до страница в същия сайт или до сайт на другия край на света.

Web – сървър е софтуерна платформа, която съдържа и обслужва различни сайтове. Потребителите, които инсталират своите сайтове на един сървър не се интересуват от неговата физическа същност или географско положение, стига този сървър да удовлетворява техните изисквания за лесен достъп до него (скорост на трансфер, обем на файловете, удобства за ползване от евентуалните посетители на сайта и др.)

Често се налага да се търси информация по даден въпрос, но с неизвестен адрес на страниците, в които се съдържа тя. В подобен случай трябва да се използват така наречените "машини" за търсене. Те са многобройни и с твърде различни възможности поради прилаганите за целта подходи и механизми. По принцип те събират предварителна информация за съществуващите Web-сайта и организират тези данни така, че при поискване бързо да могат да предоставят адреса им на потребителя. Най-популярни са **Google**, Alta Vista, Yahoo, HotBot, Lycos, dir.bg и др.

Торент е начин за паралелно търсене на определен файл (програма, филм, музика) чрез система от свързани компютри. За целта на компютъра на потребителя, който търси файла, се инсталира **торент-клиент** от типа на **Bitcomet**.

Процесът на прехвърляне на файлове с данни (хипертекстов или друг документ) от Web-сървъра към компютъра на клиента, се нарича "изтегляне" (**download**), за целта се използва така наречения HTTP-клиент (или Web-клиент), известен още като Web-браузър. Обратния процес се нарича "качване" (**upload**). Протоколът HTTP (Hyper Text Transfer Protocol) се използва главно за анонимно "сваляне" на HTML-файлове от Web-сървър (тези файлове имат разширение **.html** или **.htm**). HTML е базов WWW файлов формат.

Документите в WWW могат да бъдат пасивни и активни, а пасивните-статични и динамични. **Статичният документ** има точно съдържание . Клиентът може да вземе само негово копие,но не може да го промени.

Динамичният документ не съществува в предварително определен формат, а се създава винаги, когато някой браузър изиска документа. Когато пристигне заявка, веб-сървърът стартира приложна програма, която създава динамичния документ. Тъй като се създава нов документ за всяка заявка, съдържанието на документа може да варира за различните заявки. Динамичните документи се създават, когато се извикват от сървъра, но много приложения се нуждаят от програмата, за да тръгнат при клиента.Например това е когато програмата се извиква за да създаде анимации на екрана.

Аплетът е програма, която се изпълнява в рамките на друга програма. Java програмите, изпълнявани от Web браузъри, се наричат аплети, а не приложения, тъй като не могат да бъдат стартирани извън прозореца на браузъра.

Активни страници се разработват обработка на информация от база данни в сървърите, по заявки от клиентите. Примери за такива сайтове са портали за регистрация, формуляри за поръчка, форуми и др. Тяхното създаване е значително по-трудно.

Браузърът е програмен продукт, който позволява на потребителя да контактува по достъпен и атрактивен начин с WWW-ресурсите на Internet. Той осигурява интерфейс за навигация в Internet, свързване с отдалечени компютри, отваряне и прехвърляне на файлове, и изобразяване на текстове и изображения.

Web-браузърът изисква HTML-файла от Web-сървъра, прочита набора от HTML-команди и ги представя във вид на текст, изображения, видео и др. върху екрана на клиентския компютър.

Най-разпространените браузъри са Netscape Communicator, MS Internet Explorer и Opera. Те непрекъснато се усъвършенствуват с възможности да интерпретират създадени с нови средства кодове на Web-страници. Съгласно препоръките на консорциума W3C, съвременните браузери трябва да поддържат редица стандарти за Web-програмиране, като HTML, CSS, средства за създаване на динамични страници DOM и ECMA Script (обикновено това е Java Script), и др. В развитието си Web-браузърите следват повишаващите се изисквания на потребителите на информация в Internet. В последните си версии те предлагат мултимедийни възможности - интерпретация на аудио, видео и анимирани графични компоненти. Мобилността и повишената ефективност на WWW клиентите се подсигурава от възможност за включване на допълнителни програмни компоненти - **Plug-Ins**. Те позволиха реализация на по-сложни задачи. Почти всички браузъри включват вече възможност за разпознаване на **Java- аплети**.

4.2 Web- технологии и Web- програмиране

Всички средства за създаване на Web сайтове могат да бъдат обобщени с термина **Web технологии**. **Web- програмирането** е дейност по създаването на програмните файлове на сайта с използване на различни Web- технологии.

Приложният софтуер за Internet се базира на модела "клиент-сървър" за обслужване на потребителите. Според този модел е разделен на софтуер за клиента (компютъра, който търси информация) и за сървъра, т.е. обслужващия компютър. Клиентският софтуер превежда запитването на клиента в разбираем за сървъра вид и прави връзка с него. Сървърният софтуер управлява информационни ресурси на сървъра и обслужва насочените към него заявки.

Най-популярните Web технологии могат да бъдат условно разделени на няколко групи:

1. Технологии за програмиране на клиентската страна (client-side технологии)- HTML/XHTML, CSS, JavaScript, WML, XML и др.
2. Технологии за програмиране на сървъра (server-side технологии)- PHP, MySQL, ASP, CGI – Perl, и др.
3. Инструменти и технологии за Web мултимедия: Macromedia Flash, VRML, Macromedia Fireworks, Adobe Photoshop и др.
4. Инструменти за създаване на страници- Macromedia Dreamweaver – професионален редактор за HTML, MS FrontPage – редактор от средна класа, UltraDev и др.

Езикът HyperText Markup Language (HTML) представлява език за писане на скриптове, който описва форматирането на текста в Web страницата. Изходният код на Web-страниците е на този език. HTML е разработен първоначално от Тим Бънърс-Лий в

CERN. След това езикът преминава през няколко основни преработки. Познаването на синтаксиса на езика HTML и неговите възможности позволяват разработването на Web-страници чрез директно изписване на операторите с използване на текстообработваща програма. На създаденият текстов файл трябва да се даде разширение **.html** или **.htm**.

Въпреки, че HTML периодично бива стандартизиран, той продължава да нараства в резултат на новите тагове, атрибути и атрибутни стойности, въвеждани от разработчиците на браузъри. HTML не е език за програмиране (липсват оператори за цикли, преходи), а по-скоро редактор. Той съдържа прост набор от служебни думи (тагове) за описание на Web-документ, наречен още HTML-документ. Тагът е инструкция, заключена в начупени скоби, например <HTML> е таг, определящ началото на HTML-документа. В по-голямата си част етикетите изискват съответни затварящи етикети, предхождани от “/”, например </HTML> означава край на документа. Най-голямо достойнство на HTML е, че той позволява вмъкване в тях на хипервръзки, както и на мултимедийни елементи – графики, снимки, анимации, видео и аудиоклипове. Той е хардуерно и софтуерно независим, т.е. е един и същ за различните класове и марки компютри и операционни системи.

Каскадните набори от стилове (CSS) се състоят от специални HTML кодове, които могат да формират целия HTML документ наведнъж, като използват това, което се нарича правила. Наборите от стилове предлагат възможност за контролиране на начина, по който HTML елементите се разполагат и визуализират.

Специализираните програмни продукти значително ускоряват и улесняват създаването на Web-страници. Докато при пряко използване на HTML се въвеждат от клавиатурата команди и други условия, чрез тези програми се задават с мишката посредством менюта или икони. Освен това се предоставят шаблони (templates) за основните видове страници. Популярни представители на специализираните програми за създаване на Web-страници са Front Page, Dreamweaver, Flash и др. Те дават наистина много големи възможности и са предназначени главно за професионални дизайнери на страници.

Macromedia Dreamweaver е професионален, но сравнително лесен за използване HTML редактор- средство за създаване на авторски материали и за управление на Web-сайтове. Dreamweaver е редактор от вида **WYSWYG** (What you see is what you get това, което виждате, е това, което получавате). С него могат да се създават и нови сайтове, и да се редактират съществуващи.

Macromedia Flash е програма за генериране на компактни векторни графики и анимации за WWW. Под управлението на компонента за динамичен сървър на Flash - Generator, филмите на програмата могат директно да контактуват със скриптове и програми от страна на сървъра. Могат да бъдат кодирани звуци във формата MP3, с което се постига разпространяване във WWW с високо качество при минимален размер на файловете.

4.3 Web- дизайн

Изкуството за създаване на общия стил на Web-сайта и художественото му оформяне е **Web дизайн**. Дизайнът на Web страниците не е точна наука, но основите му могат да се усвоят на базата на спазването на прости правила и указания, както и при наблюдаване на успешно или зле направени сайтове. Счита се, че добрият дизайнер трябва да има завършено художествено образование и да натрупа опит чрез продължителна практика по разработване на сайтове.

Качеството на Web-сайта несъмнено се определя най-вече от качеството на съдържащата се в него информация. От значение за качеството са и критерии, като

удобство на ползване и прегледност на представяне на информацията, наличие на подходящи за навигация хипервръзки, валидност на дадените хипервръзки и оформлението и атрактивността на страниците.

Разработването на несложни Web-сайтове вече е масово практикуван и доста рутинен процес, състоящ се от следните етапи:

- а) подготовка за създаване на Web-сайта, изразяваща се в следното:
 - изясняване на целите – към кого е адресирана и какво трябва да се постигне с информацията в сайта;
 - планиране на общото информационно съдържание на сайта и разбиването му по страници;
 - планиране на вътрешните хипервръзки и връзките с външни сайтове;
 - съставяне на текста за всяка страница;
 - подготвяне на мултимедийните обекти(графики, снимки, аудиофайлове и др.), които са предвидени за отделните страници;
- б) реализиране на отделни Web-страници чрез достъпните за целта програмни средства;

в) проверка на готовите страници и съхраняването им като файлове;

г) прехвърляне (инсталиране) на файловете върху Web-сървър.

За да бъде атрактивна и с висока посещаемост, освен богато информационно съдържание, Web-сайта трябва да включва следните компоненти- вътрешни и външни хипервръзки, фон на текстовете, бутонни икони, бутони за връзка с e-mail на разработчика на Web- сайта, фотографии, възможности за увеличаване на мащаба на включените снимки, анимирани изображения (собствени или заимствани от други сайтове), промяна на фона (транспарентни изображения), чувствителни карти, формуляр за обратна връзка с разработчика на сайта, таблици, прозорци и рамки, внедрени аудио- и видеофайлове и работещ брояч на посетителите на страницата. Важно е и общото впечатление от стила на сайта като цяло.

В настоящият момент при разработването на един сериозен Web-сайт участват четири групи специалисти.

- Web-мениджъри - осигуряват общата организация на работата;
- Web-дизайнери - определят стила на сайта;
- Web-програмисти - разработват HTML-кода на сайта;
- специалисти по рекламата - определят мястото на рекламните материали в сайта;

При създаване на Web-сайт трябва да се търси максимално опростена и прегледна структура. При липса на опит и въображение, се препоръчва преглеждането на чужди Web-страници за идеи.

Основните компоненти на Web-сайт са:

-**текст**- особенното при текста за Интернет е кодирането му на съответния език на текста, например за кирилицата кодиранията са Windows-1251(най-използваното за Windows), KOI-8(руска), ISO(международен стандарт) и др. За правилното възпроизвеждане се указва съответната кодова таблица (Encoding). Неправилно зададеното кодиране може да направи страницата нечетима за много потребители дори на вашия компютър тя да изглежда добре;

- **изображения** (снимки, рисунки, знаци) - те са обикновено във форматите GIF и JPEG. Пълноцветни изображения обикновено се оформят като JPEG файлове, знаци, рисунки, текстове - като GIF (могат да са прозрачни участаци, да се зареждат прогресивно на стъпки). Други изображения могат да се включват като файлове, изобразяването им обаче зависи от софтуера на клиента;

- **фон** (Background) на страницата - може да е едноцветен (определен цвят) или изображение. Ако е изображение то се размножава толкова на брой пъти колкото е голяма видимата област от прозореца на браузъра за извеждане на документа;

- **таблици** - служат за разполагане на елементи на страницата върху екрана - текстове, изображения, компоненти. Таблиците могат да бъдат произволно разделяни и обединявани, да имат рамки, фонове, да се разполагат на екрана във фиксирани размери или в процентно съотношение. Последното позволява да се съобразява разположението на елементите с най-разнообразните монитори на клиентите - с коренно различна големина и разделителна способност;

- **анимация** - най-простия вид са няколко изображения от (например от тип GIF) редуващи се едно след друго в определен интервал от време;

- **звук** - може да бъде фон (звучи по време на четене на отворена страница) изреждаем - възпроизвежда се от специални програми, които зависят от типа на звуковия файл.

- **видео (мултимедия)** - последователност от кадри и звуков съпровод. Съществуват редица формати за съхранение на видео, за да се възпроизведат те браузъра на клиента се нуждае от програми (players), наречени plug-ins (добавки).

- **файлове(за download)** - с произволно съдържание и формат: документи, музика, филми, програми и т.н.

При дизайна трябва да се имат предвид следните елементи и идеи:

- да се използват възможностите на фреймовете и таблиците при оформяне на страниците;

- да се прилагат текстови стилове или стилови макети;

- анимациите трябва да се използват с мярка, за да не изглеждат страниците претрупани;

- да се внимава как се разполагат текста и изображенията, като се спазват принципите за добро оформление;

- да се използва единен стил в цветовете на фона и в изображенията на страницата;

- да се записват най-добре оформените страници във вид на шаблони, за да се използват след време като база за създаване на други страници;

- много е важно да се провери HTML-кода на сайта и да се тества готовия сайт чрез поне 2 различни браузъра.

При дизайна могат да се използват безплатни сайтове, от които да се вземат различни обекти, които да се включват в разработваната страница.

Ефективен е всеки сайт, който изпълнява предназначението си. Ако обаче никой не посещава този сайт, той е неефективен. От опитните разработчици на сайтове могат да се заимстват редица правила, спазването на които предотвратяват това. Могат да се посочат някои най-общии примери.

Самата начална страница трябва ясно да заявява каква е целта на сайта. Изображенията не бива да засенчват съдържанието и да го правят нечетливо. Да се използват кратки страници (не по-дълги от 1,5 екрана) с конкретно съдържание, които се зареждат бързо и от които потребителите получават желаната информация, а не разтеглени страници, съдържащи информация по различни теми. Броят на различните шрифтове и основни цветове да не надхвърля 3. Хипервръзките до конкретната информация не трябва да превишават 3 нива и др.

Много доставчици по света предлагат безплатна услуга **инсталиране** на клиентски Web - страници на различни сървъри. На английски език този термин е **Free web-hosting**. Трябва да се има предвид, че тези услуги се спонсорират от рекламодатели и почти винаги безплатно инсталираните страници на Web- сайтове са

оградени от редица рекламни банери, менюта и др., което не винаги е желано и не съответства на съдържанието и стила на сайтовете. Обикновено доставчиците ограничават:

Домейн- имената на Web-сайтовете. При безплатното инсталиране те са поддомейни, например не се дават имена от типа **zhelanie.net** , а **zhelanie.freewebspace.com**. Многото и дълги имена затрудняват бързото набиране и стартиране на търсения сайт.

Максимално допустимия общ обем на файловете в сайта се ограничава до 50 MB, което не винаги е приемливо за клиента.

Максимален размер на отделен файл- обикновено до 299-300 KB. Не винаги позволява инсталирането на големи снимки, аудио- и видео- файлове.

Максимален месечен трафик- обикновено 2-3 GB.

Цените на платените услуги за Web -инсталирането от български оператори никак не са високи, затова си струва да се прецени дали не е по-добре срещу минимално заплащане да се инсталира сайт без всякакви ограничения.

4.4 Правила за етична и безопасна работа в Интернет

Етичните въпроси, свързани с използването на Интернет, могат да се разгледат в следните направления:

- опити за несанкциониран достъп до ресурси (т.н. “ хакване”);
- използване на източниците за информация със спазване на авторското право;
- сигурност и защита на личните данни;
- използване на мрежата за неморална и престъпна дейност.

Сигурността при ползването на Интернет следва да се обмисли и оцени в две основни направления:

- Сигурност при достъпа до системата
- Сигурност при информационния обмен в мрежата

Политиката на сигурност е система от правила в дадена организация, която обхваща мерки най-малко в следните области:

- Физическа защита на отделните работни места и на мрежата;
- Организационни мерки, определяне на правата и задълженията на всекиго;
- Политика на достъп до ресурсите (нива на достъп, пароли);
- Мониторинг и санкции за нарушенията на политиката на сигурност;
- Използване на системи за кодиране на информацията , и др.

Широкото разпространение на информационните технологии е разширило възможностите за събиране, обработка и използване на личните данни. В условията на новите технологии, човек оставя “следи”, например като използва кредитни карти и мобилни телефони, по които движението му може да бъде проследено, чрез специално създадени глобални системи.

Европейският парламент обръща внимание, доколко такива системи работят в полза на сигурността и каква е възможността за прехвърлянето на личните данни на проследеното лице към недоброжелателни хора. Установени са дори случаи на използване за индустриален шпионаж. Преразглеждането на тези системи има за цел вземането на достатъчно ефективни мерки срещу злоупотребите и при борбата със световния тероризъм.

Друга опасност за личната неприкосновеност в интернет са програмите, които събират лична информация за потребителите, понякога без тяхното знание. Тези данни могат да бъдат използвани във вреда на потребителите.

Типичен метод за събиране на информацията са т.нар. **cookies** (бисквитки) – данни за движението на един потребител, събирани на сървърите за техните посетители. Те се събират главно за технологични цели и могат да бъдат забранени от всеки потребител. ввропейските институции са за забрана на тази технология, с която според тях се нарушава Европейската конвенция за правата на човека, предвид това, че от данните на потребителя, лесно се стига да спам и други нежелани ефекти. При използването на cookies трябва да се иска изричното съгласие на потребителите, а анонимното ползване на съдържанието да бъде разрешено.

Правила за работа с лични данни в Интернет.

1. Да се предупреждават потребителите, относно информационната политика на компанията или институцията, преди от тях да бъде събрана персонална информация.
2. Да се предоставя възможност за избор на потребителите да решат дали и как да предоставят исканата от тях информация.
3. Да се предоставя на потребителите достъп до събраната информация.
4. Да се гарантира използването на събраната информация в съответствие с оповестената политика за защита от неоторизирана употреба.
5. Контрол върху изпълнението на правилата за защита на данните. Налагане на санкции случаи на нарушаване на правилата.
6. Да не се предоставят личните данни – име, домашен дрес, име на училището, телефонен номер в кореспонденция чрез Интернет. Дори данни, които са смятани за недостатъчни за идентификация, могат да послужат за точно описание на семейство, адрес, приятели.

При ползване на обществени системи не се препоръчват действия, които:

- поставят комуникационната система в нарушение на закона;
 - претоварват комуникационната система;
- създават допълнителни разходи при лично ползване

Правила за ползване на електронна поща:

- придържане към вътрешни правила за електронна поща и носене на лична отговорност;
- да се докладва всяко подозрение за нарушение на политиката за електронна поща на администратора;
- всеки файл, получен чрез електронна поща да бъде проверен с антивирусни програми;
- да се спазват правилата за копиране и защита на интелектуалната собственост;
- да се пазят паролите и кодовете за достъп.

Действия, които представляват неприемливо използване на електронната поща:

- изпращане на авторски материали;
- изпращане на електронни писма, които се приемат за клеветнически, нападателни, сплашващи, оскърбителни или обидни;
- използване на чужда идентификация или парола, без съответното оправомощаване;
- причиняване на намалена функционалност на мрежата;
- публикуване, съхраняване, преработване на материали на неприличен спрямо общоприетите принципи език;
- разглеждане и разпространение на поверителна информация;
- повреждане, изтриване или блокиране на достъпа до файлове за общо ползване;

- инсталиране, копиране, съхранение или използване на софтуер, в нарушение на законодателството за защита на интелектуалната собственост и правилата за лицензиране.

Основните понятия на **авторското право** имат други значения в информационното общество и особено в Интернет. В **Мрежата** има привидност за пълна свобода, всичко изглежда възможно и разрешено, но не е така. Копирането на чужд материал е забранено, колкото издаването на книги без съгласието на автора. Освен това има и нови категории нарушения, например предоставяне на технически средства за достъп и употреба на защитен материал.

При закупуване на компютърни програми, потребителят не плаща цена за придобиване на авторските права, те обикновено са за компаниите-производители. Потребителят няма право да копира, препродава или изменя тези програми по силата на т.нар. лицензия. За компютърното пиратство се предвижда и наказателна отговорност. При подаден сигнал до полицията, се извършва проверка на място с разрешение на съда. При наличие на нелицензиран софтуер, носителите и компютрите се изземват.

Според лицензионните споразумения, потребителите нямат право:

- да инсталират и използват компютърни програми без необходимата за това лицензия;
- да възпроизвеждат компютърни програми в един или повече екземпляри, по какъвто и да е начин и под каквато и да е форма с изключение на резервно копие;
- да разпространяват компютърните програми;
- да преработват или внасят промени в компютърните програми или в техния програмен код;
- да предлагат достъп по безжичен път или чрез кабел или друго техническо средство до компютърните програми.

В Интернет се предлагат програми, които се разпространяват свободно и зареждането им от Мрежата и разпространението им не е нарушение на авторското право. В лицензиите на някои компютърни програми изрично се допуска възпроизвеждането, разпространението или изменението им.

Един от проблемите в Интернет е неспазването на авторските права от недобронамерени потребители, които копират изображения от сайтовете.

Цифрово създадените водни знаци е техника, която позволява на автора скрито да добавя информация за авторското си право и други съобщения за проверка на цифрови звукозаписи, видео или изображения, сигнализируют също и документи. Това е скрито съобщение – група битове, описващи информация, имаща отношение към сигнала или автора на сигнала (наименование, местоположение и т.н.). Техниката взема това название от създадените водни знаци на книга или банкноти като мярка за безопасност. Цифрово създадените водни знаци не са форма на стеганографията, в която данните са скрити в съобщенията без знанието на крайния потребител, въпреки че много методи за правене на водни знаци имат стеганографската особеност, че не могат да бъдат забелязани с човешко око. Комерсиалните **стеганографски програми** се концентрират върху прибавянето на дигитални водни знаци към аудио файлове и изображения. За разлика от стеганографията, водните знаци внедряват допълнителна информация във всяка част на изображението. Тези знаци се разчитат от специални програми и съдържат данни кога е бил създаден файла, кой държи авторските права, начин за връзка с автора и т.н.

ТЕСТ

1. Хипервръзките са предназначени за:

- а) достъп само до други програми от типа "браузър";
- б) достъп само до различни части на web-страницата;
- в) достъп само до други web-сайтове;
- г) достъп само до различни части в web-страницата и до други web-сайтове;

2. На кой език за програмиране е написан изходния код на хипертекстов документ?

- а) PASCAL;
- б) COBOL;
- в) C++;
- г) HTML.

3. Кражбата на пари през интернет е:

- а) кибертероризъм
- б) компютърно престъпление
- в) престъпление извършено чрез компютърни технологии
- г) нито едно.

4. Политика на сигурност е:

- а) множество правила ,осигуряващи защита
- б) набор от средства за техническа и програмна защита
- в) политика на достъп до ресурсите
- г) средства за мониторинг на системата за защита.

5. Стеганографска система е:

- а) вид криптографска система
- б) вид комуникационна система
- в) система за предаване на скрита информация
- г) вид система за контрол на достъпа.

6. Изключете излишното:

- а) Dreamweaver;
- б) Opera;
- в) FrontPage;
- г) UltraDev.

7. Изключете излишното:

- а) Dreamweaver;
- б) Opera;
- в) Internet Explorer;
- г) Netscape Navigator.

8. Bitcomet е названието на

- а) web-технология;
- б) торент-клиент;
- в) e-mail - клиент;
- г) ftp- клиент.

9. Системата FrontPage се използва за:

- а) Web-дизайн;
- б) текстообработка;

- в) сканиране за сайтове;
- г) редактиране на Linux-приложения.

10. При дизайна на сайт броят на използваните видовете шрифтове не трябва да превишава:

- а) 5;
- б) 4;
- в) 6;
- г) 3.

Л И Т Е Р А Т У Р А

1. Бърнев, П., Керпеджиев, С., Основни понятия в информатиката, С., 1988
2. Боровска П., Компютърни системи, Сиела, С., 2003
3. Ганчев И., Компютърни мрежи и комуникации, Пловдив, 1999.
4. Колев К., Станев С., Ганчев И., Информатиката, Шумен, 2000.
5. Манев К., Манева Н., Информационни технологии за 9 и 10 клас, С., 2000.
6. Тянев Д. С., Електронни цифрови машини, ВМЕИ-Варна, 1995.
7. Уайт Р., Как работят компютрите (илюстрации Тимоти Даунс), 1999.
8. Нишева, М., Д. Шишков, Изкуствен Интелект, Интеграл, Добрич, 1995.
9. Ненков, Н. Експертни системи, Ш., 2006
10. Станев, Ст., Смит, П., Захариев, Ф., Компютърни системи и мрежи, Ш., 2002
11. Тотков, Г., и др., Основи на компютърната информатика, Пловдив, 2000
12. Йорданова Г., Буров, И., Приложен софтуер, Шумен, 2004.
13. Иванов, И., Кънчева, С., Информационни технологии, С., 2000.
14. Цонев, И., Милев, П., Желязков, К. Информационни системи и сигурност, Ш., 2005
15. Станев, С., Железов, С., Компютърна и мрежова сигурност, Ш., 2005.

Приложения

Приложение 1

Пробразуване на числата

След като се работи с повече от една бройна система, е редно да се въведат означения за това, в каква система е даденото число. В противен случай има опасност от грешки. Така например 11 може да означава 11 в десетичната система, 3 – в двоичната и 17 – в шестнадесетичната. Най-разбираемо означение е изписване на основата на бройната система вдясно като индекс: $11_{(10)}$, $11_{(2)}$, $11_{(16)}$. То затруднява обаче отпечатването при регистрация на данните. За означаване на данните в

шестнадесетична система по-често се използва буквата Н (от Hexadecimal - шестнадесетичен), която се записва непосредствено след числото – 11Н. За означаване на двоичната бройна система се използва В.

Десетичната, двоичната и шестнадесетичната бройни системи са позиционни. Това означава, че стойността, която изразява цифрата, зависи от нейното място (позиция) в числото.

В общия случай числото X може да се представи в еднородна позиционна система с основа p по следния начин:

$$X = x_{n-1} \cdot p^{n-1} + x_{n-2} \cdot p^{n-2} + \dots + x_1 \cdot p^1 + x_0 \cdot p^0 + x_{-1} \cdot p^{-1} + \dots + x_{-m} \cdot p^{-m} = \sum_{i=-m}^n x_i \cdot q^i, \text{ където:}$$

i – номер на текущия разред

m – брой дробни разреди

q – основа на бройната система

x_i – разредни коефициенти, съставлящи “азбуката” на дадената бройна система.

Десетичната бройна система има десет различни цифри - от 0 до 9, чрез които могат да се представят всички числа. Всяко число, по-голямо от 9, може да бъде представено чрез комбинация от тези цифри. Стойността, която изразява цифрата зависи от позицията, имаща определено “тегло”, например:

| 4 | 3 | 2 | 1 | Позиция |
|---------------|--------------|-------------|------------|---------|
| $10^3 = 1000$ | $10^2 = 100$ | $10^1 = 10$ | $10^0 = 1$ | Тегло |

Броя на цифрите в числото определя броя на неговите разреди. Разредът с най-малко тегло се нарича младши разред, обикновено той е най-десния на числото. Например за числото $1999_{(10)}$:

| 4 | 3 | 2 | 1 | Позиция |
|--------|--------|--------|--------|---------|
| 10^3 | 10^2 | 10^1 | 10^0 | Тегло |
| 1 | 9 | 9 | 9 | |

$$1999_{(10)} = 1 \cdot 10^3 + 9 \cdot 10^2 + 9 \cdot 10^1 + 9 \cdot 10^0$$

Двоичната бройна система има за основа числото 2, и само 2 цифри – 1 и 0. Позиционните тегла се базират на степените на основата – 2.

| 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Позиция |
|-----------|----------|----------|----------|---------|---------|---------|---------|---------|
| $2^7=128$ | $2^6=64$ | $2^5=32$ | $2^4=16$ | $2^3=8$ | $2^2=4$ | $2^1=2$ | $2^0=1$ | Тегло |

За да се преобразува ръчно десетично число в двоично, се използва правилото за деление на цялата част и умножение на дробната част с основата на бройната система, в която трябва да се получи резултатът. Цялата част на изходното число се дели, докато се получи 0. Последният остатък от делението е най-старшият разред на числото в новия код, а първият остатък – най-младия. При преобразуването на дробната част умножението продължава, докато се получи 0 (което не винаги е възможно) или докато се запълнят необходимият брой разреди.

Пример: Да се преобразува ръчно $13,625_{(10)}$ в двоично число.

Цялата част е $13_{(10)}$, а дробната – $0,625_{(10)}$

Преобразуване (ръчно) на цялата част:

$$13 : 2$$

$$\underline{12} \quad 6 : 2 \quad 13_{(10)} = 1101_{(2)}$$

$$1 \quad \underline{6} \quad 3 : 2$$

$$0 \quad \underline{2} \quad 1$$

$$1$$

Преобразуване на дробната част:

$$0,625$$

$$X \quad 2$$

$\overline{1}$, 250 пренос 1 (най-старши бит)
 X 2
 $\overline{0}$, 500 пренос 0
 X 2
 $\overline{1}$ 000 пренос 1 и умножението (най-младши бит)
 $0,625_{(10)} = 0,101_{(2)}$
 цялото число $13,625_{(10)} = 1101,101_{(2)}$

Преобразуването на двоично в десетично число ръчно става с помощта на посочената по-горе формула, като най-напред намират разредите, като този в ляво от запетаята получава номер 0.

Пример: $1100,11_{(2)} = 1.2^3 + 1.2^2 + 0.2^1 + 0.2^0 + 1.2^{-1} + 1.2^{-2} = 12 + 0,5 + 0,25 = 12,75_{(10)}$

За правила на преобразуването могат да се използват и следните съотношения:

$$1_{(10)} = 1_{(2)} = 1.2^0$$

$$2_{(10)} = 10_{(2)} = 1.2^1$$

$$4_{(10)} = 100_{(2)} = 1.2^2$$

$$8_{(10)} = 1000_{(2)} = 1.2^3$$

$$16_{(10)} = 10000_{(2)} = 1.2^4$$

Броят на нулите след единицата е равен на степента на 2.

Шестнадесетичният код се използва главно за съкращаване представянето на многоразредни двоични числа, когато с тях трябва да програмира човек, а не компютър. Основата на тази система е числото 16. То не е избрано произволно – 16 е броят всички комбинации на едно четириразредно двоично число. В шестнадесетичната бройна система се използват цифрите 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, и буквите от латинската азбука A, B, C, D, E, F, като с буквите се означават десетичните числа 10, 11, 12, 13, 14 и 15. В табл. В.1 е дадена връзката между числа в шестнадесетична и двоична бройна система.

Една шестнадесетична цифра се представя с 4 двоични разреда.

Шестнадесетичното число лесно може да се преобразува в двоично чрез замяна на всяка шестнадесетична цифра със съответната ѝ група от четири двоични цифри (наречена тетрада) от Таблица 1

| | | |
|------|------|------|
| 6 | B | 3 |
| 0110 | 1011 | 0011 |

$$6B3_{(16)} = 11010110011_{(2)}$$

Табл.1

| <i>Двоично число</i> | <i>Шестнадесетично число</i> |
|----------------------|------------------------------|
| 0000 | 0 |
| 0001 | 1 |
| 0010 | 2 |
| 0011 | 3 |
| 0100 | 4 |
| 0101 | 5 |
| 0110 | 6 |
| 0111 | 7 |
| 1000 | 8 |
| 1001 | 9 |

| | |
|------|---|
| 1010 | A |
| 1011 | B |
| 1100 | C |
| 1101 | D |
| 1110 | E |
| 1111 | F |

За да се преобразува число от двоичен в шестнадесетичен код, двоичното число се разделя на групи от по 4 разряда (бита), започвайки отдясно наляво. След това всяка група от 4 разряда (тетрада) се заменя с шестнадесетичния и еквивалент. Ако най-старшата група има по-малко от 4 разряда, тя се допълва с нули отляво (когато числото е цяло, а не дробно).

Пример: Да се преобразува двоичното число

11001011101000,1110110 в шестнадесетичен код.

Разделя се числото на групи от по 4 разряда.

| | | | | |
|------------|------|------|-----------|------------|
| 11 | 0010 | 1110 | 1000,1110 | 110 |
| 0011 | | | | 1100 |
| 3 | 2 | E | 8,E | C |
| най-старши | | | | най-младши |
| разред | | | | разред |

За преобразуването на числа от десетична в шестнадесетична система може да се използва описаната по-горе процедура за преобразуване на десетични в двоични числа, като се има предвид, че основата на бройната система е 16.

В компютрите не се използват непозиционни бройни системи. Често за пример за такава система се дава римската бройна система, макар че строго погледнато, тя не е "чисто" непозиционна.

Например 1999 година, написана в римската система е MCMXCIX, а 2000 година – MM.

Логически функции.

Инструкциите и данните в микрокомпютрите се кодират в двоичен код с две значения 0 и 1, които се срещат като логическа 0 и логическа 1, ниво 0 и ниво 1, сигнал 0 и сигнал 1 или само 0 и 1. Когато нивото на логическата 0 е по-ниско от нивото на логическата 1, кодирането се нарича положителна логика.

При две променливи A и B съществуват 16 различни логически функции и всяка от тях може да се представи чрез трите логически функции И (AND), ИЛИ (OR) и НЕ (NOT). Връзката между входните променливи (аргументи) и изходните величини на една логическа функция или схема може да се изрази чрез таблица на истинност и чрез логическо уравнение.

Таблицата на истинността се състои от две части. Първата част съдържа всички възможни състояния на входните величини, а втората част определя изходните състояния на логическата схема като функция на съответната комбинация от състоянията на входните величини.

Таблиците на истинност на основните логически функции И, ИЛИ, НЕ и функцията с две променливи са следните:

| <u>НЕ</u> | | <u>И</u> | | | <u>ИЛИ</u> | | | изключващо ИЛИ (сума по модул 2) | | |
|-----------|---------------|----------|---|------------------|------------|---|----------------|-------------------------------------|---|------------------|
| X | $Y = \bar{X}$ | X | Y | $Z = X \wedge Y$ | X | Y | $Z = X \vee Y$ | X | Y | $Z = X \oplus Y$ |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| | | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|

Операцията НЕ се реализира в компютрите с логически елемент, наречен инвертор, който има един вход и един изход. Сигналят на изхода на инвертора е винаги отрицание (инверсия) на входния. Ако входния сигнал е 1, изходният е 0 и обратно.

Логическа функция И (конюнкция, логическо умножение) се реализира с логически елемент И, който има два или повече входа и един изход. На изхода на елемента И има сигнал 1 само когато на всички входове са подадени едновременно 1. Ако само на един от входовете се подаде 0, сигналят на изхода става 0.

Логическа функция ИЛИ (дизюнкция, логическо сумиране) се реализира с логически елемент ИЛИ, който има два или повече входа и един изход. На изхода на елемента ИЛИ има 0 само когато на всички входове е подаден сигнал 0. Ако само на един от входовете се подаде 1, сигналят на изхода също става 1.

Освен функциите И и ИЛИ се използват и техните комбинации с инвертиране НЕ. Схемата И-НЕ (NAND) е комбинация на схемата И и инвертор.

Често се използва и логическата функция изключващо ИЛИ, наричана сума по модул 2 или логическа неравнозначност. Тази функция има на изхода си 1 само ако двете променливи А и В са различни.

Елементите И – НЕ, реализирани по интегрална технология, са в основата на такива възли на компютрите като регистри, броячи, суматори и други.

Логическите функции се използват при проектирането на основните устройства на хардуера на компютрите и при реализирането на логическите операции при работата на АЛЮ.

Приложение 2

ASCII- кодова таблица

| Bit Positions 3210 | Bit positions 654 | | | | | | | |
|-----------------------|-------------------|-----|-------|-----|-----|-----|-----|-----|
| | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| 0000 | NUL | DLE | SPACE | 0 | @ | P | ' | P |
| 0001 | SOH | DC1 | ! | 1 | A | Q | a | q |
| 0010 | STH | DC2 | " | 2 | B | R | b | r |
| 0011 | ETX | DC3 | # | 3 | C | S | c | s |
| 0100 | EOT | DC4 | \$ | 4 | D | T | d | t |
| 0101 | ENQ | NAK | % | 5 | E | U | e | u |
| 0110 | ACK | SYN | & | 6 | F | V | f | v |
| 0111 | BEL | ETB | ' | 7 | G | W | g | w |
| 1000 | BS | CAN | (| 8 | H | X | h | x |
| 1001 | HT | EM |) | 9 | I | Y | i | y |

| | | | | | | | | |
|------|----|-----|---|---|---|---|---|-----|
| 1010 | LF | SUB | * | : | J | Z | j | z |
| 1011 | VT | ESC | + | ; | K | [| k | { |
| 1100 | FF | FS | , | < | L | \ | l | |
| 1101 | CR | GS | - | = | M |] | m | } |
| 1110 | SO | RS | . | > | N | ^ | n | ~ |
| 1111 | SI | US | / | ? | O | _ | o | DEL |