ПЛОВДИВСКИ УНИВЕРСИТЕТ „ПАИСИЙ ХИЛЕНДАРСКИ“
ФАКУЛТЕТ ПО МАТЕМАТИКА, ИНФОРМАТИКА И ИТ
КАТЕДРА „КОМПЮТЪРНИ СИСТЕМИ“

КОМПЮТЪРНИ МРЕЖИ И КОМУНИКАЦИИ

Изготвил: гл. ас. д-р Генчо Стоицов

Пловдив, 2013 г.

УВОД

Телекомуникационният сектор е област на висок растеж в европейската икономика и важен неин стимулатор. Този ръст на развитие е продиктуван от нарастващата необходимост от по-бързи и качествени услуги. Това се постига основно чрез промяна на технологиите, особено по отношение на цифровизацията и преносимостта на данни. Динамичното му развитие задължава повишаването на познавателната активност на обикновения потребител.

Тематичното съдържание на учебната дисциплина обхваща следните теми:

Лекции

- Общи понятия свързани с телекомуникациите. Комуникационни системи, линии, канали, мрежи, подмрежи. Компютърни мрежи. Мултиплексиране. Маршрутизация и комутация;
- Комуникационен модел OSI за съединение на отворени системи. Основна идея. Нива. Основни понятия - услуга, интерфейс, протокол. Протоколи-характеристики, функции. Елементи за стандартизация;
- Локални компютърни мрежи (LAN). Физически слой в LAN – топология, кабелна система. Канален слой в LAN. Международни стандарти за физическия и каналния слой в LAN;
- Шумоустойчиво кодиране. Непрекъснати и блокови кодове. Режими на използване на шумоустойчивите кодове. Примерни кодове;
- Шифриране на данните. Симетрично и асиметрично шифриране. Хеш функции;
- Междумрежови комуникации. Съгласуване на мрежи в долните слоеве (internetworking). Устройства за междумрежови комуникации-повторители, концентратори, модеми, мостове, комутатори, маршрутизатори, шлюзове;
- Комуникационен модел TCP/IP. Комуникационни слоеве. Адресация в модела TCP/IP. Маршрутизация с помощта на IP адреси – протоколи;

Лабораторни упражнения

1. Стандарт IEEE 802.3 и неговите разновидности. Физически компоненти за изграждане на локална мрежа. Кабелна система. Конектори. Междинни устройства;
2. IEEE 802.11 (Wi-Fi) мрежи. Логическа архитектура. Разновидности на стандарта. Режими на работа. Сигурност;
3. Радио-параметри на безжичната среда;
4. Междумрежови комуникации. Междинни устройства;
5. Комуникационен модел TCP/IP. Адресиране;
6. Задачи за калкулиране на подмрежови маски, префикси, подмрежи и адреси на устройства от мрежата;
7. Протоколи;
8. Шумоустойчиво кодиране. Линейни и циклични кодове;
9. Прости виртуални топологии за локални мрежи;
10. Виртуални топологии с рутиране.

На студентите са предоставени:

- електронен вариант на лекционния курс;

  Адрес: **http://kmk.fmi-plovdiv.org/LecturesKMK.pdf**

- динамични имитационни модели (комуникационни симулакруми) разработени на Flash, показващи функционирането на реални комуникационни процеси;

  Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/**

  User: student
  Pass: Student-111

- фърмуери на устройства достъпни на адреси:
  http://www.tp-link.com/en/support/emulators/
  http://www.dlink.com/support/faq/?prod_id=1457
  http://ui.linksys.com/files/
- задачи за формиране на оценка по упражненията

  Адрес: **http://kmk.fmi-plovdiv.org/Zadachi.pdf**

- допълнителна информация

  Адрес: **http://kmk.fmi-plovdiv.org**

1. Основни понятия използвани в темите от раздела

1.1. Основни понятия свързани с физическото разпространение на сигнала

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=12**

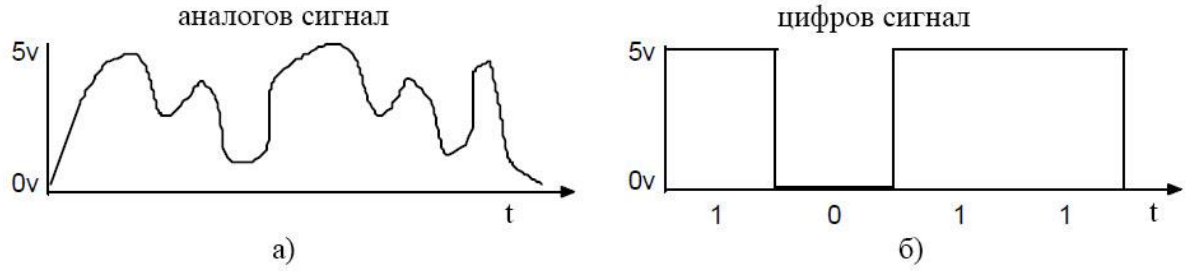
При разглеждане на темите засягащи компютърни мрежи и комуникации е необходимо да се уточнят някои основни понятия участващи в тях. Тези понятия имат основно значение за разбирането и усвояването на представения материал.

Информация – съвкупност от сведения за каквито и да са събития, явления и предмети.

Съобщение – начин на предаване на информацията. То се предава с помощта на сигнал.

Сигнал – изменяща се физическа величина (електрически ток, звукова вълна и т.н.), еднозначно изобразяваща съобщението. Делят се на два вида:

- *аналогови* – има безкраен брой значения за ограничен интервал от време. На фигура 1а е показан сигнал, където волтажът варира между 0 и 5 волта.



фигура 1 *Аналогов и цифров сигнал*

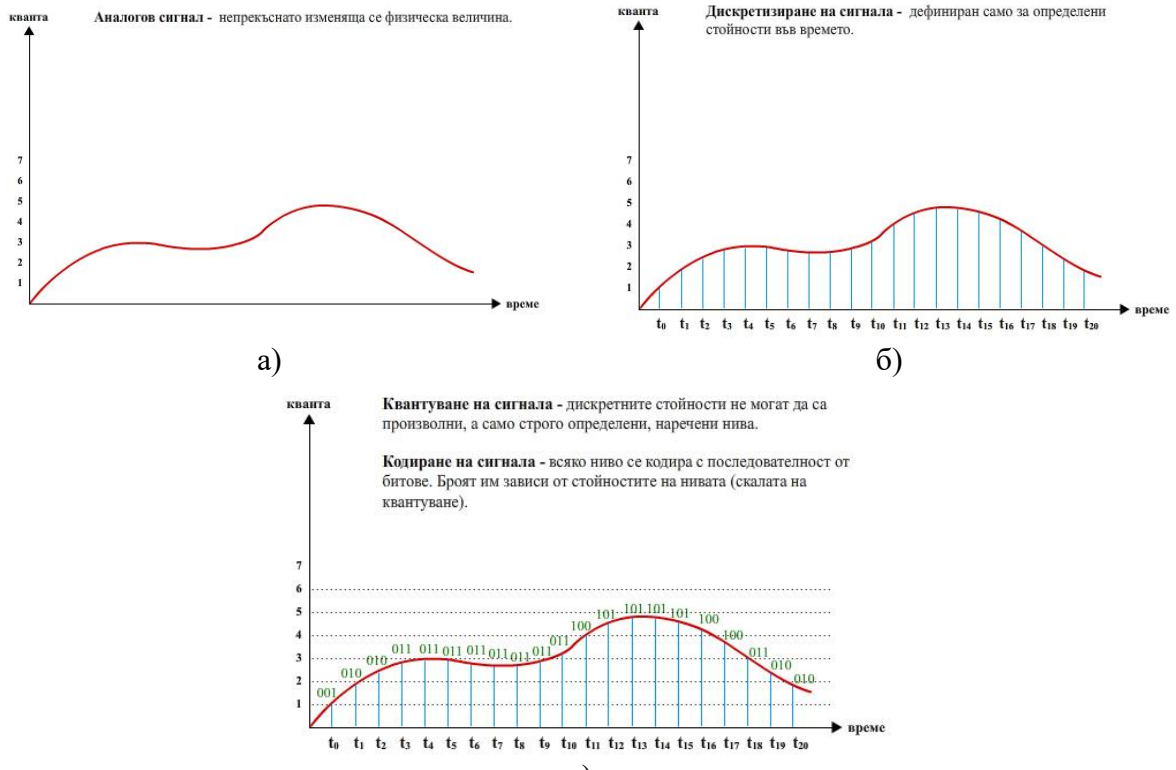
Едни от основните предимства на аналоговите сигнали са, че могат лесно да бъдат мултиплексирани и са по-неподатливи към затихването.

- *цифрови* - има краен брой значения за ограничен интервал от време. На фигура 1б символичните означения за цифровия сигнал са 0 и 1, а техните електрически значения са съответно 0 и 5 волта.

Цифровият сигнал е дискретен сигнал, който се отличава с допълнителна характеристика, наречена квантуване.

Най-простият математически модел на дискретния сигнал (фигура 2б) представя краен брой значения $\mathbf{s_i=s(t_i)}$, където $\mathbf{i=1..n}$. Стъпката на дискретизация $\mathbf{\Delta t=t_{i+1}-t_i}$ е постоянна. Този тип сигнал е дискретен по оста на времето.

При квантуването последователността от отчети $\mathbf{s_i}$ са строго определени и се наричат нива (фигура 2в). Разликата между две съседни нива се нарича "стъпка на квантуване", а разликата между дискретната стойност и присвоеното ниво - "грешка от квантуване" или още "квантов шум".



фигура 2 *Фази на цифровизация на сигнала*

Разликата между дискретния и цифровия сигнал може да се демонстрира със следния пример:

Пример 1. Нека дискретният сигнал има следните стойности: u_0=3.2257[V], u_1=3.1252[V], u_2=3.5255[V], u_3=3.4252[V] …, а стъпката на

квантуване по ниво е Δu=0.1[V]. Тогава стойностите на цифровия сигнал ще бъдат u[0]=3.2[V], u[1]=3.1[V], u[2]=3.5[V], u[3]=3.4[V] ….

Представените фази от фигура 2 са последователни етапи от процес наречен импулсно-кодова модулация (Pulse Code Modulation) и е с най-голямо практическо приложение. Тя лежи в основата на цифровите преносни системи и модулирането на телефонни съобщения.

Цифровото предаване има някои предимства пред аналоговото:

- оборудването за цифрово предаване е по-евтино;
- цифровите сигнали са по-малко уязвими към грешки, предизвикани от смущения.

Превръщането на съобщението в сигнал се състои от следните операции, които могат да бъдат независими или съвместни. Това са:

- *преобразуване* – превръщане на неелектрическите величини, съответстващи на първоначалното съобщение, в електрически сигнал;
- *кодиране* – построяване на сигнала по определен принцип, имащ някакъв математически израз (ASCII, EBCDIC). Кодирането обхваща превръщането на аналогово или цифрово съобщение в цифров сигнал. Устройството, което се използва в този случай се нарича кодек и обединява функциите кодиране и декодиране на сигнала (фигура 3);

фигура 3 *Кодиране на сигнала*

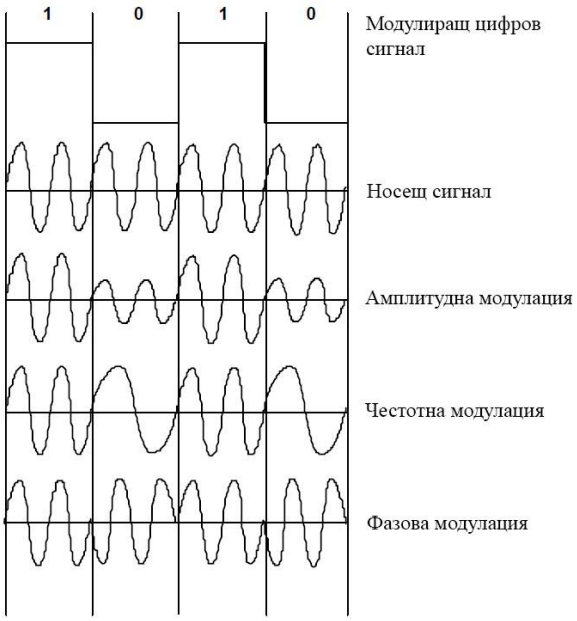
- *модулация* – въздействие върху някой параметър на електрическия ток (амплитуда, честота, фаза), благодарение на което в изменението на този параметър се оказва заложен предавания сигнал (фигура 5). Модулацията обхваща преобразуването на аналогово или цифрово съобщение в аналогов сигнал. Използва се за прехвърляне на цифрови данни през аналогова линия. Устройството, което се използва се нарича модем (modem – modulator/demodulator).

фигура 4 *Модем*

фигура 5 *Модулиране на сигнал*

На фигура 4 е изобразено преобразуването на цифров в аналогов сигнал и обратно. Възможните типове модулация (фигура 6) са:

- *Амплитудна модулация* (AM - Amplitude Modulation) – амплитудата представлява силата на сигнала от гледна точка на височината на вълната. При този модулация амплитудата на носещия сигнал се променя спрямо стойностите на битовете от модулиращия цифров сигнал. Например, две стойности на амплитудата (малка и голяма) могат да съответстват на 0 и 1. Този тип модулация е податлив на изкривяване;
- *Честотна модулация* (FM - Frequency Modulation) – честотата представлява времето, за което вълната изпълнява пълния си цикъл. При тази модулация честотата на носещия сигнал се променя спрямо стойностите на битовете от модулиращия цифров сигнал. Например, две стойности на честотата (ниска и висока) могат да съответстват на 0 и 1. FM е по-устойчива на изкривяване от AM;
- *Фазова модулация* (PM - Phase Modulation) – фазата е относителното състояние на една вълна спрямо друга, измерена в градуси. При тази модулация фазата на носещия сигнал се променя спрямо битовата стойност на модулиращия цифров сигнал. Промяната на фазата на носещия сигнал показва смяна на стойностите от 0 към 1 и обратно на модулиращия сигнал.

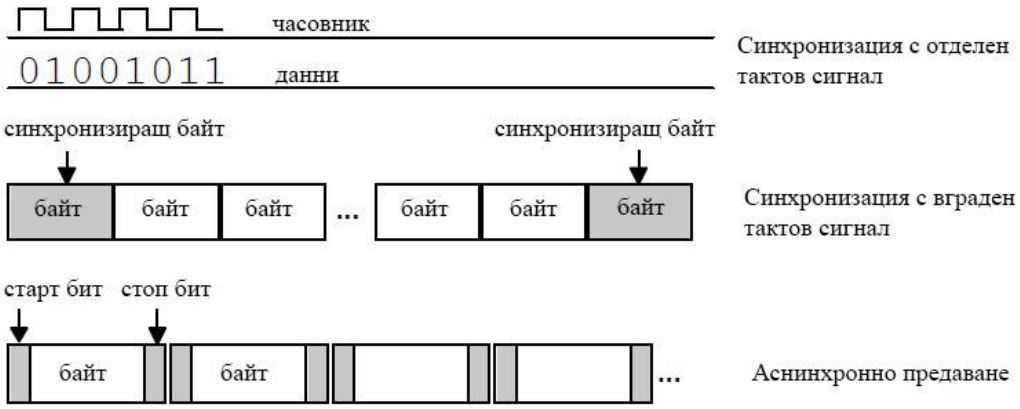


фигура 6 *Видове модулация*

За правилното интерпретиране на сигналите при комуникацията е необходимо предавателят да уведоми приемащото устройство кога да очаква получаването на данни. Това позволява на приемника да се подготви за тази процедура. Уведомленията трябва да са достатъчно често, за да се поддържа комуникацията във времето. Този подход се нарича синхронизация.

Съществуват два основни метода на синхронизация: *синхронно* и *асинхронно* предаване. При синхронното предаване (фигура 7) съществува вграден механизъм за тактуване, координиращ предавателя и приемника. Той може да бъде реализиран чрез отделен тактов сигнал или тактовата информация да бъде вградена в сигнала с данните. Информацията се представя като непрекъсната последователност от символи, групирани по блокове (кадри) с относително голям размер.

Асинхронното предаване използва старт и стоп бит (фигура 7) за начало и край на всяко съобщение, за да може приемащото устройство чрез него да синхронизира своя вътрешен тактов генератор с този на предавателя.



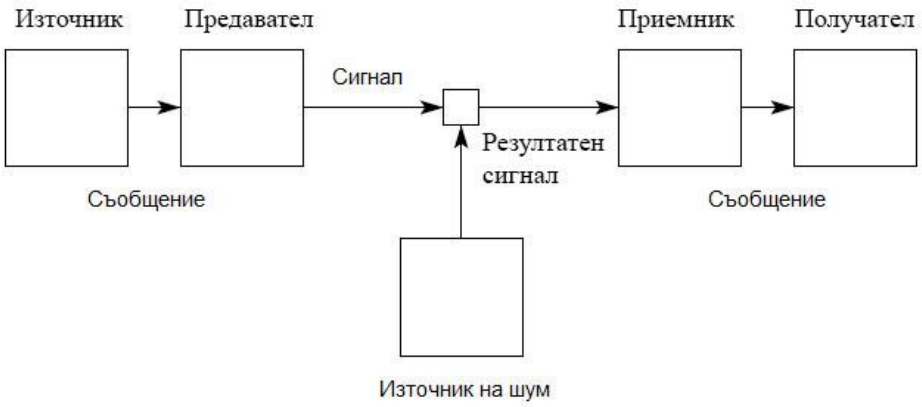
фигура 7 *Методи за синхронизация*

1.2.Основни понятия свързани с комуникационните системи

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=13**

За предаването на едно съобщение на разстояние е необходима комуникационна система.

Комуникационна система – съвкупност от технически средства, необходими за предаване на съобщения от източника към получателя. Това са: предавател, комуникационна линия и приемник.



фигура 8 *Общ вид на комуникационна система*

Според вида на предаваните съобщения [94] комуникационните системи се делят на:

- телефонни (за предаване на глас);
- телеграфни и телетексни (за предаване на текст);
- факсимилни (за предаване на неподвижни изображения);

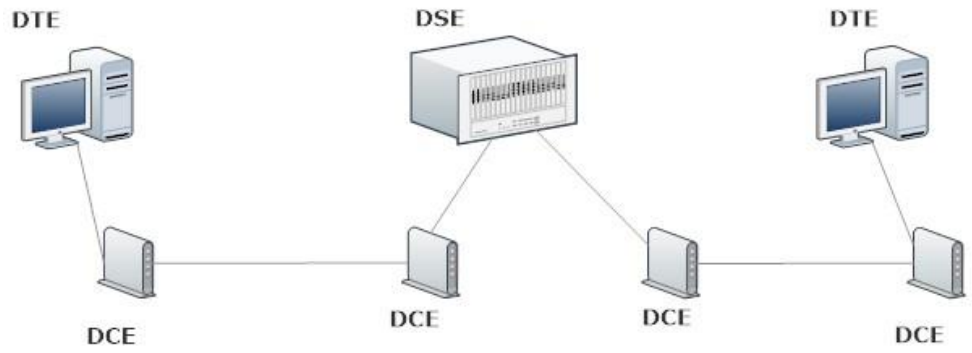
- телевизионни и видеотелефонни (за предаване на подвижни изображения);
- телеизмервателни (за предаване на данни от измервания на разстояние);
- системи за предаване на данни.

Единицата за измерване на количеството информация е *бит* (b).

Количеството информация, което може да се предаде по дадена комуникационна система за единица време определя нейната пропускателна способност. Единицата за измерване на пропускателната способност е бит за секунда (b/s).

Оборудването за мрежи (фигура 9) може да се класифицира в три големи категории:

- *Data Terminal Equipment (DTE)* – това са устройствата на потребителите, които преобразуват изходящите потребителски данни в трансмисионен сигнал и обратно, входящия сигнал в потребителски данни. DTE може да бъдат: терминали, терминални адаптери, персонални компютри, mainframe компютърни системи и др.;
- *Data Circuit-terminating Equipment (DCE)* – мрежово оборудване позволяващо към него да бъде свързано DTE устройство за използване на комуникационната линия. Може да бъде реализирано като част от DTE устройството или като самостоятелна единица. Неговите функции се свързват с установяване, поддържане и прекратяване на сесията в мрежата, както и с конвертиране на сигналите за предаване. Модемите и мултиплексорите са примери за DCE устройства. Например, при серийни комуникации DCE страната определя скоростта на връзката (позната като clocking), затова при настройки на сериен интерфейс се задава т.нар. clock rate. DTE устройството се подчинява на тези настройки (CISCO – *Router(config-if)#clock rate 64000*);
- *Data Switching Equipment (DSE)* – мрежово оборудване свързващо DCE устройства и осигуряващо комутационни възможности към мрежата. Отговорни са за преноса на данните в мрежата. Най-често се наричат комутатори (switch).



фигура 9 *Оборудване за мрежи*

Комуникационна линия – физическа среда, която се използва за предаване на сигналите от предавателя към приемника.

Комуникационната линия е *физическо* понятие.

Трябва да се прави разлика между *физическо* и *логическо* понятие. Дадено понятие е *логическо*, когато обхваща съвкупност от средства, които определят типа му, докато с термина *физическо* понятие назоваваме единичен физически обект.

Комуникационен канал – съвкупност от средства, осигуряващи предаване на сигнал от някаква точка А на комуникационната система до друга нейна точка Б.

Комуникационният канал е *логическо* понятие.

В зависимост от вида на сигналите (цифрови или аналогови), постъпващи на входа и излизащи на изхода на канала, каналите [94] се делят на:

- аналогови;
- цифрови;
- аналогово-цифрови;
- цифрово-аналогови.

Най-често под канал се разбира логическа част от използваната физическа комуникационна линия, осигуряваща предаването на отделен сигнал. На фигура 10 може да се определи всеки от горепосочените видове канали. Например, за канала маркиран от точките А и В може да се каже, че е цифрово-аналогов, а каналът AD – цифров.

фигура 10 *Схема на връзка между два компютъра, чрез модем и телефонна линия*

Комуникационните канали се характеризират със следните особености:

- наличие на шум в канала, даже при отсъствие на полезен сигнал в него;
- линейност;
- закъснение на сигналите;
- затихване на сигналите;
- деформиране на сигналите.

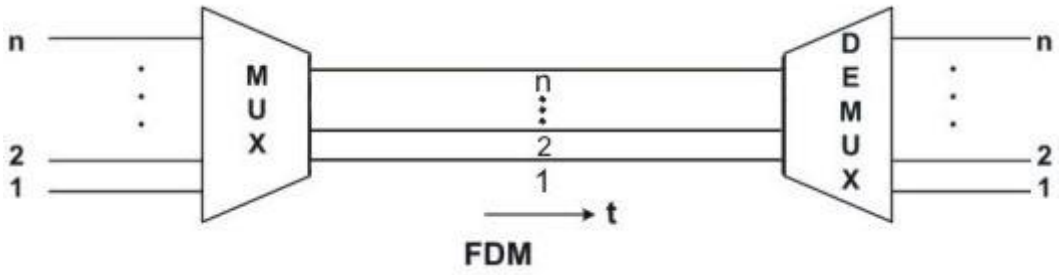
Една комуникационна система се нарича *многоканална*, ако осигурява няколко паралелни канала за предаване по една обща комуникационна линия. За целта се използват устройства наречени: мултиплексор (MUX) и демултиплексор (DEMUX).

фигура 11 *Мултиплексиране*

На фигура 11 е показана възможността за обединяване на няколко (n на брой) входни комуникационни канала за пренос през обща среда с висок капацитет.

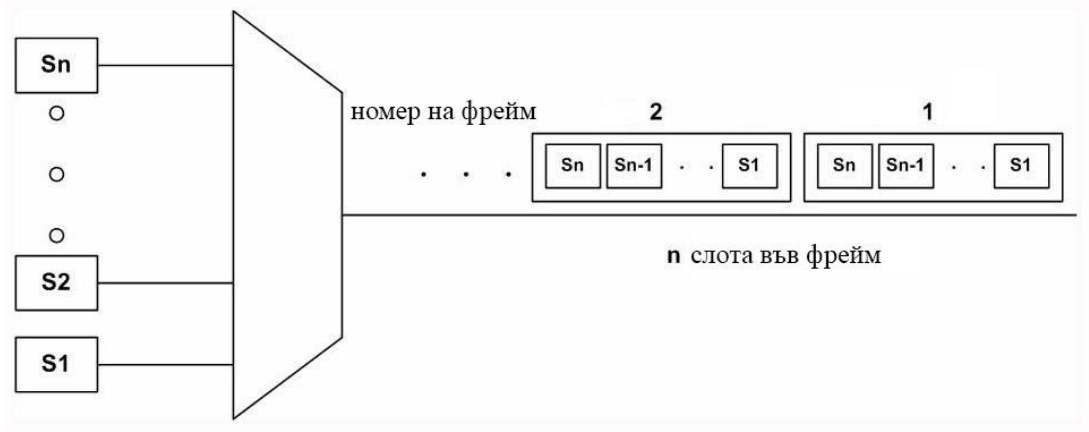
При многоканалните комуникации се използват няколко способа за разделяне на каналите:

- *честотно деление* (FDM - Frequency Division Multiplexing) – разделя честотната лента на няколко независими честотни канала за предаване (фигура 12). По между им има защитни честотни ленти за предпазване от смесването на сигнала. Използва се главно в радио и телевизионните комуникационни системи;



фигура 12 *Честотно деление (FDM)*

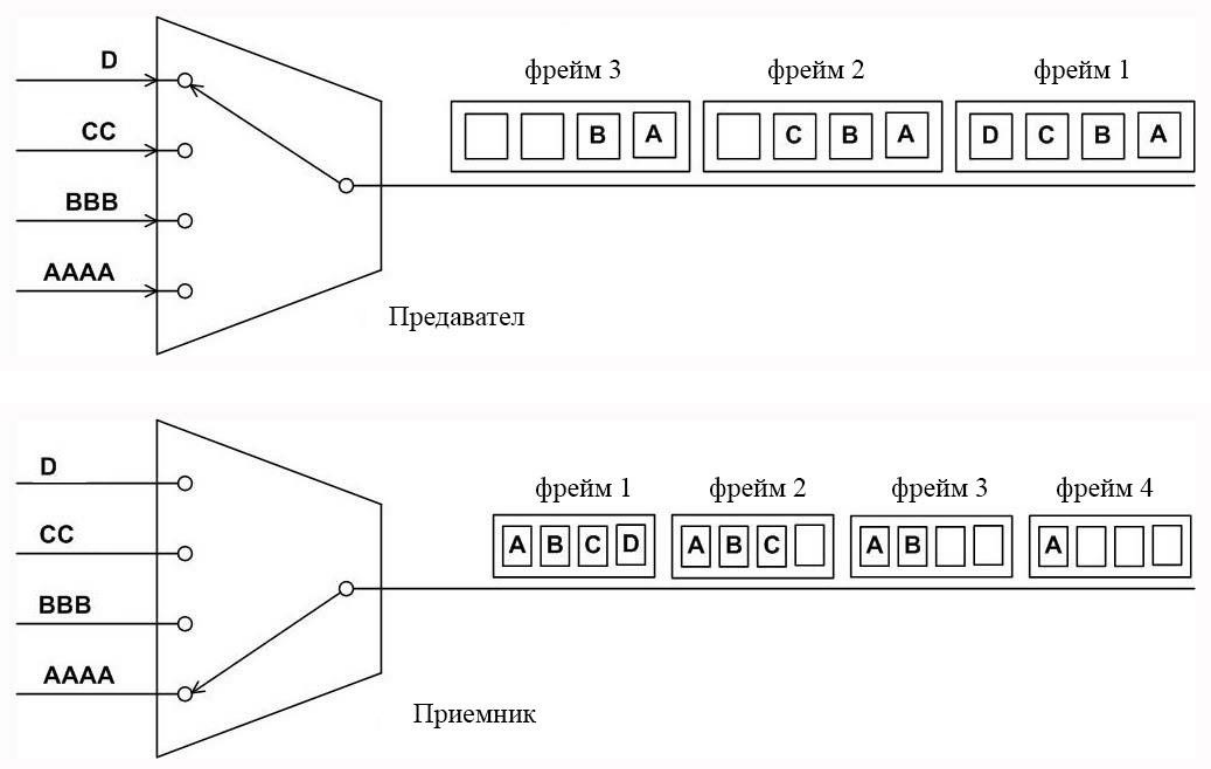
- *времеделение* (TDM – Time Division Multiplexing) – за всеки канал се пада определен интервал от време (периодично повтарящ се), през който той предава. Честотната област е една и съща. Използва се основно в телефонните комуникационни системи. Електронен комутатор комбинира входните данни в съставен сигнал, който преминава по трасето и се демултиплексира от съответния комутатор от приемащата страна. Входните данни се буферират на малки части. Всеки буфер е с размер бит или символ. Буферите се сканират последователно и достатъчно бързо се освобождават, за да се образува съставният поток от данни без да се възпрепятстват пристигащите данни. Скоростта в изходната линия трябва да бъде по-голям или равен на сумата от отделните скорости на входните данни. Съставният сигнал може да бъде пренесен директно или чрез модем.



фигура 13 *Времеделение (TDM)*

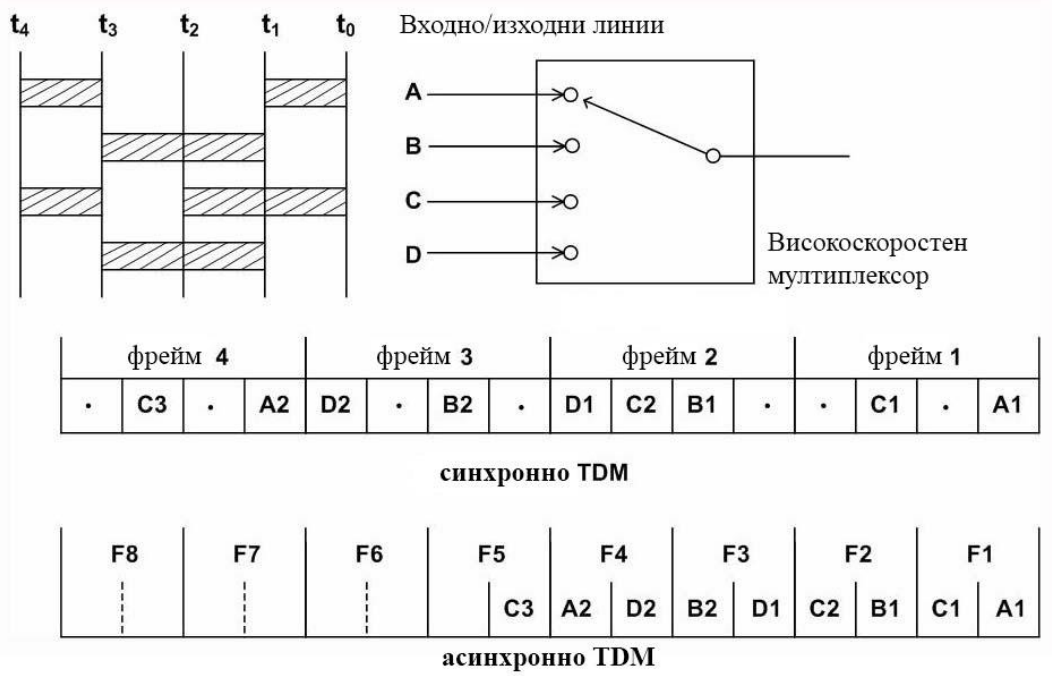
На фигура 13 се вижда, че при формирането на фреймовете последователно се включват слотове от различните източници. Всеки слот S_i е предварително определен за конкретен източник, независимо от това

дали източниците имат данни за изпращане или не (фигура 14). Този тип мултиплексиране се нарича още синхронно TDM;



фигура 14 *Мултиплексиране и демултиплексиране при TDM*

- *статистическо деление* (SM – Statistical Multiplexing) – подобно на времеделението с тази разлика, че ако времето определено за конкретен канал не се използва от него се заема от следващия. Използва се в стандартите за глобални компютърни мрежи (X.25, Frame Relay, ATM). Нарича се още асинхронно или интелигентно TDM. Мултиплексорът сканира входните буфери и запълва с данните текущия фрейм, след което го изпраща (фигура 15). Демултиплексорът приема фрейма и разпределя данните към съответните буфери. В този случай данните в слотовете трябва да притежават и адресна част за идентифициране на източника им (фигура 16).

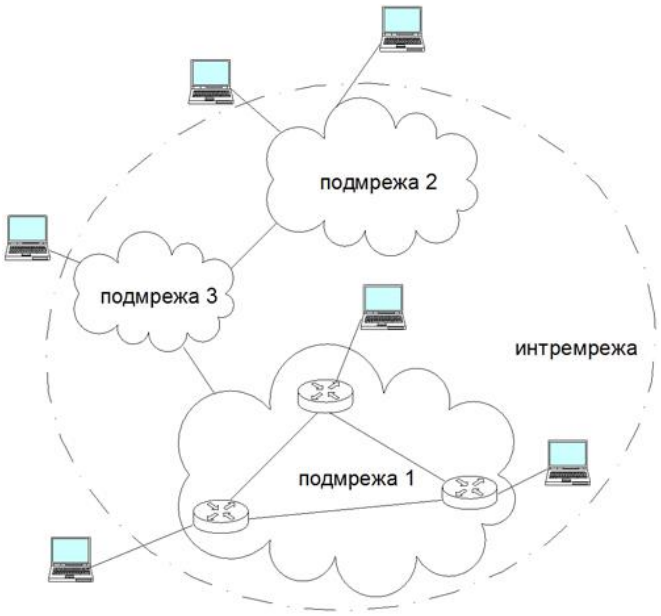


фигура 15 *Съпоставка между TDM и SD*

фигура 16 *Адресиране на слот при SD*

Комуникационна мрежа – съвкупност от различни комуникиращи устройства свързани помежду си чрез комуникационни линии (фигура 17).

Комуникационна подмрежа – съвкупност от комуникационни линии и междинни мрежови възли (комутатори/маршрутизатори), осигуряващи предаването на информация между крайните възли (фигура 17). Крайните възли не се включват в подмрежата.



фигура 17 *Комбинирана схема за визуално представяне на понятията комуникационна мрежа и подмрежа, интермрежа и междинен мрежов възел*

1.3.Основни понятия свързани с компютърните мрежи

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=14**

Компютърна мрежа – частен случай на комуникационната мрежа, чиито крайни възли са главно компютърни системи (фигура 17).

Комуникационна интермрежа – съвкупност от взаимосвързани комуникационни мрежи (фигура 17).

За правилното предаване на съобщение по мрежата се грижат междинните мрежови възли (маршрутизатори/комутатори), изпълняващи две основни функции: маршрутизация и комутация.

Маршрутизация – процесът на намиране на оптимален маршрут за преминаване на дадено съобщение по мрежата.

Комутация – процесът на пренасочване на съобщението от даден входен порт на междинния мрежов възел към определен негов изходен порт, водещ към следващия междинен възел от избрания маршрут.

Различните мрежи използват различни методи за комутация:

- *комутация на канали* - комутацията се извършва на три етапа: установяване на временен канал между източника и получателя, обмен на съобщения, разпадане на канала;
- *комутация на съобщения* - съобщението се предава по различните участъци на мрежата с натрупване в междинните й възли. Изисква повече буферна памет в маршрутизаторите, които да съхраняват дългите съобщения, докато се освободи изходяща линия;
- *комутация на пакети* – при възела подател съобщението се разделя на по-малки части (пакети - от 1500 байта до 8000+ при бързите мрежи > 1Gbps), всеки със своята адресна и информационна част. Отделните пакети се предават между комутаторите на подмрежата, докато накрая достигнат до възела-получател. Съществуват различни режими на работа:

- *дейтаграмен режим* – при този режим пакетите (дейтаграми) се снабдяват с пълни адресни заглавия, по които комутаторите на пакети определят по-нататъшния им маршрут. Отделните дейтаграми се предават независимо една от друга, затова маршрутите им могат да са различни и може да бъде нарушен техния първоначален ред. За целта на отделните дейтаграми се дават номера, по които в крайния визел-получател се извършва подреждане в първоначалния им ред. Прилагат се в приложения, които не изискват висока надеждност на комуникациите, а обмен на къси съобщения – SNMP например;
- *режим на виртуално съединение* – при този режим между взаимодействащите крайни възли предварително се изгражда логическо съединение, по което след това се предават всички пакети на съобщението един след друг по реда на следване, след което съединението се разпада. Използва се от приложения, които изискват надеждност на връзката-www, ftp, e-mail и др.

2. Темата за относителния OSI стандарт

2.1.Общи бележки

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=15**

Съвременните мрежови архитектури следват принципите на модела OSI (Open Systems Interconnection). Създаден е от Международната организация по стандартизация ISO (International Standards Organization) за връзка между отворени системи. Отворена система е система, чиито ресурси могат да се използват от другите системи в мрежата.

Този модел категоризира различните процеси на комуникационната сесия на различни функционални нива. Нивата са организирани спрямо естествената поредица от събития, възникващи по време на комуникацията. Именно това разграничаване позволява категоризацията на различните междинни устройства от компютърната подмрежа. По този начин по-лесно може да бъде обяснена и разбрана тяхната роля и функционалност.

2.2.На какво да се обърне внимание

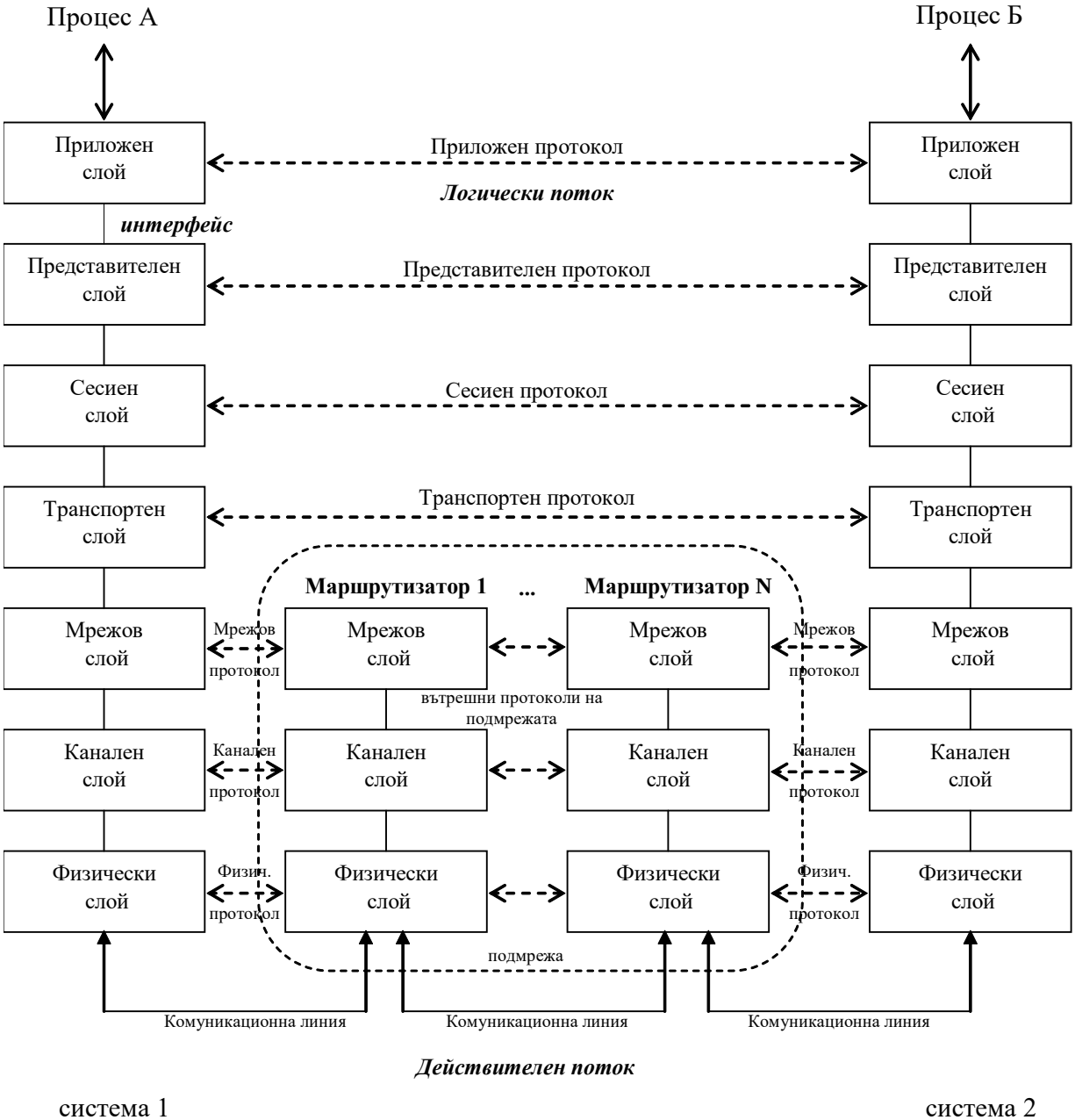
Когато се разглежда структурата и значението на OSI модела е необходимо да се обърне внимание на следното:

- архитектура - нивата и тяхната функционалност;
- основни понятия – услуга, интерфейс и протокол;
- елементи за стандартизация – спецификация на протокола, дефиниция на услугата и адресация.

Относителният OSI модел е съставен от седем нива с различна функционалност, следваща естествения път на комуникация между два процеса. Всяко едно от тях се състои от обекти, изпълнява определена логическа функция и предлага специфични *услуги* за по-горния слой. Съвкупността от правила за взаимодействие между обекти от едноименни слоеве се нарича *протокол*, а правилата за взаимодействие на обектите от съседни слоеве на една и съща система се нарича *интерфейс*.

Във всеки слой има три елемента на стандартизация споменати по-горе. Първият от тях е *спецификация на протокола*, което означава да е ясна структурата на неговата единица за данни, семантиката на всичките и

полета, начина на предаването ѝ и т.н. *Дефиниция на услугата* определя услугите предоставени за по-горния слой. Тук трябва да се отбележи, че самият модел не казва как да бъде направено това. *Адресацията* се състои в идентифициране на точките за достъп (SAP-Service Access Point) до предлаганите услуги от конкретния слой.



фигура 18 *Комуникационен сценарий на модела OSI*

Когато се говори за комуникация между два обекта от едно и също ниво трябва да се прави разлика между логически и действителен поток на данните. Действителният поток преминава вертикално по нивата, докато за

логически се счита непряката комуникация между два обекта от един и същи слой.

Повечето съвременни мрежови модели се различават по степента, в която са наследили OSI модела. Често нивата се свиват до по-малко на брой, което означава и преразпределяне на функционалността. Въпреки това всеки един от тях може да се съпостави с OSI модела и да се посочат нивата, до които той се разполага. Самата комуникация може да използва само част от нивата. Като пример може да се даде функционирането на единичен LAN сегмент, където обмена може да се извърши до второ ниво.

2.3.Кратко описание на нивата на относителния OSI стандарт

Нивата на OSI, както беше отбелязано по-горе са седем на брой и са разположени вертикално (фигура 18).

2.3.1. **Физическо слой** *(physical layer)*

Физическото ниво е най-долния слой на стандарта. Той е непосредствено свързан с комуникационната линия. Основната му функция е предаване на неструктуриран поток от битове по нея. На това място трябва да отбележим, че под понятието *комуникационна линия* ще разбираме физическата среда, която се използва за предаване на сигналите. Физическото ниво е ограничено до процесите и механизмите, необходими за предаването на сигналите в средата за разпространението им. То не включва самата среда, но задава спецификациите за производителността и. Съществуват четири функционални области:

- механична-задава броя и дължината на проводниците между устройствата, формите и размерите на конекторите и др.;
- електрическа – определя формите, продължителността и нивата на сигналите;
- функционална–обуславя смисловото значение на електрическите сигнали, които си разменят съседни системи;
- процедурна–специфицира последователността от събития, чрез която се обменя потока от битове между два обекта на физическия слой.

На това ниво няма механизъм за определяне на значението на предаваните битове, което означава, че не може да се определи тяхната валидност. Основни функции са:

- изграждане и разпадане на физически съединения;
- преобразуване на съобщенията в сигнали;
- физическо предаване на битове;
- синхронизация по битове;
- реализиране на физическия интерфейс;
- диагностика на определен клас неизправности.

Устройствата, които осигуряват съгласуваност между два сегмента на това ниво са: повторител(repeater) и концентратор(hub).

Повторителите са хардуерни устройства, чиито основни функции са свързани с възстановяване и усилване на сигнала, удължаване на покривното разстояние и съгласуване между сегментите във физическия слой.

Концентраторите са хардуерни устройства, осигуряващи възможност за лесно включване на допълнителни възли в локалната компютърна мрежа.

2.3.2. Канален слой *(data-link layer)*

Този слой използва услугите на физическия слой, разширява техните възможности и ги предоставя на мрежовия слой. Отговорен е за надеждното предаване на данните. Това означава осигуряване на надежден канал между два мрежови възли с отсъствието на каквито и да е грешки. Протоколната единица за данни се нарича кадър и съдържа достатъчно служебна информация за проверка на нейната правилност. Каналният слой е отговорен за откриването и коригирането на грешките на ниво кадър, както и превръщането на потоците от битове в кадри. Форматът на кадрите се определя от избрания протокол на канално ниво. Функциите на каналния слой обикновено се реализират смесено - апаратно и програмно. Колкото повече функции са реализирани софтуерно толкова по-ниска е производителността.

Протоколите от този слой се разделят на две основни категории: асинхронни и синхронни. Трябва да се отбележи, че има разлика в

понятието синхронизация за физическото и каналното ниво. На физическо ниво то се свързва с определянето на границите на битовете на базата на общ синхронизационен сигнал. На канално ниво смисълът е в разделянето на контролните от потребителските данни. Синхронните протоколи използват за граници уникална поредица от битове. Те могат да бъдат символно или битово ориентирани. При символно ориентираните протоколи потребителските данни се състоят от последователност от символи ограничени от уникални контролни символи (SYN и EOT). Този тип протоколи са символно зависими, защото се базират на определен набор от символи (например, ASCII или EBSDIC). При битово ориентираните протоколи няма специфичен набор от символи, както при символно ориентираните. Последователността от битове се предава под формата на кадър, съдържащ потребителски, контролни и адресни данни, контролна сума, флаг за начало и за край.

Пример за синхронен символно ориентиран протокол е Binary Synchronous Control (BSC; познат като BISYNC), създаден от IBM през 1960 г. за полудуплексна комуникация. Битово ориентиран протокол е High-level Data Link Control (HDLC), който се определя от стандартите ISO 3309, ISO 4335 и ISO 7809 и поддържа полудуплекс и пълен дуплекс.

Устройствата, които осигуряват съгласуваност между два сегмента на това ниво са: мост (bridge) и комутатор (switch).

Основното предназначение на един мост е препредаване и филтриране на кадрите, използвайки указаните в тях MAC адреси на възлите получатели. Използва се най-често за сегментиране на големи и претоварени локални мрежи на по-малки мрежи.

Комутаторът е концентратор с възможност за комутация на кадри в каналния слой. Използва се за намаляване на вероятността от конфликти в IEEE 802.3 мрежи с интензивен трафик.

Физическият и каналният слой са необходими за всеки тип комуникация.

2.3.3. Мрежов слой *(network layer)*

Мрежовият слой управлява функционирането на подмрежата. Тук трябва да се уточни, че понятието *подмрежа* означава съвкупност от комуникационни линии и междинни мрежови възли

(комутатори/маршрутизатори), осигуряващи предаването на информация между крайните възли. Крайните възли не се включват в подмрежата. Мрежовото ниво е отговорно за установяването на маршрут, които да се използва при комуникацията. То няма вграден механизъм за откриване и съответно коригиране на грешки при предаване. Разчита на надеждните услуги на каналния слой. Използва се за обмен на данни между системи, намиращи се в различни локални сегменти отделени чрез маршрутизатори.

Основни функции на това ниво са:

- адресация;
- маршрутизация;
- комутация;
- управление на натоварването.

Адресацията е необходима за еднозначна идентификация на адресираните обекти на мрежовия слой. Обикновено се използва йерархичен принцип на адресация, при който пълният адрес се състои от няколко степени, като първата от тях специфицира адреса на мрежата, втората–адреса на крайния възел(хоста), третата–идентификатора на виканата програма(порта). За пример може да се посочи стандартът IPv4, при който адресът е четири байтов. Йерархията е по-проста, включваща две степени: адрес на мрежата и адрес на хоста.

Маршрутизацията е най-важната функция на мрежовия слой. Свързана е с избиране на оптимален маршрут за преминаване на пакетите през подмрежата на базата на предварително зададен критерий. Протоколната единица за данни се нарича пакет. Пакетите са с фиксирана големина в рамките на една мрежа, но при извършване на преход между две мрежи е възможно пакетът да се раздели на части (фрагментира), след което да се възстанови (дефрагментира). Например, преходът LAN-WAN-LAN.

Комутацията е нужна поради факта, че липсва във всеки един момент пряко съединение между всеки два възела на подмрежата. Намира приложение и на канално ниво.

Управление на натоварването е свързано с избягването на задръствания в подмрежата, при което рязко се влошават нейните характеристики.

Маршрутизаторът (router) е устройството, което свързва хетерогенни мрежи на мрежово ниво. Той представлява отделен многопротоколен мрежови възел със собствен адрес. Използва се и за свързване на локална към глобална мрежа (например, Internet).

За маршрутизаторите този слой е последен. Функциите на мрежовия слой, както и на по-горните слоеве се реализират програмно.

2.3.4. Транспортен слой *(transport layer)*

Транспортният слой осигурява услуга, подобна на тази от каналния слой-сигурност на сегменти-извън локалната мрежа. Той може да открива липсващите или сгрешени сегменти и да изисква тяхното препредаване. Грижи се за подредбата им, ако са пристигнали в разбъркан ред, управлява транспортните съединения свързващи крайните възли, управлява съответствието между мрежови адреси и транспортни адреси.

Транспортният слой освобождава по-горния сесиен слой от грижата за надеждното и ефективно транспортиране на данните между крайните системи.

2.3.5. Сесиен слой *(session layer)*

Петото ниво на OSI модела се нарича сесийно и се използва относително рядко като самостоятелно. Повечето протоколи свързват функциите му с тези на транспортно ниво. Основната му функция е да управлява комуникационния поток, наречен сесия. Към тези функции спадат:

- управление на диалога-двупосочен едновременен диалог (full duplex - FD), двупосочен алтернативен диалог (half duplex - HD), еднопосочен диалог (simplex);
- установяване, възстановяване и прекратяване на сесия;
- работа с пароли;
- осигуряване на статистика за работата на мрежата.

На това ниво се управляват т.нар. синхронизационни точки, които при грешка в предаването позволяват сесията да бъде възстановена от последната достигната точка.

Пример за протокол от този слой е Network Basic Input Output System (NetBIOS) на Microsoft. Той създава сесия между две машини работещи под Windows операционна система, използвайки просто задаване на имена.

2.3.6. Представителен слой *(presentation layer)*

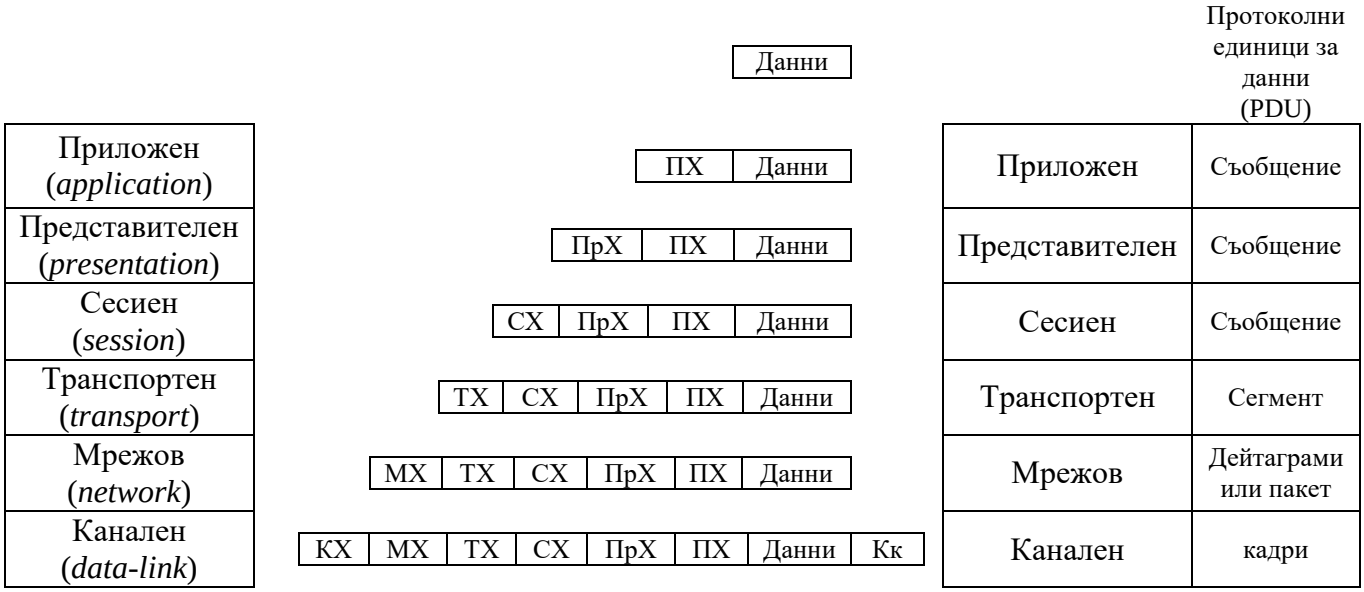
Представителното ниво е отговорно за управление кодирането на данните, свързано е със синтаксиса и семантиката на предаваните данни. Този слой е предназначен за преодоляване на различията във форматите, кодовете и структурите на данните. Осигурява кодиращи/декодиращи услуги.

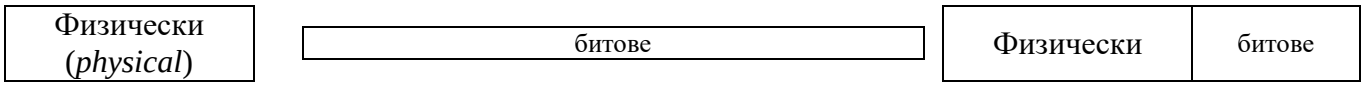
2.3.7. Приложен слой *(application layer)*

Най-високото седмо ниво на OSI модела е Приложният слой. Важното е да отбележим, че то не включва потребителските приложения, само осигурява интерфейса между тези приложения и мрежовите услуги. Това ниво осигурява услуги на приложните процеси и приложни протоколи, които ги реализират. Например:

- Достъп чрез HTTP;
- Достъп чрез FTP;
- Електронна поща;
- Файлови и принтерни услуги;
- Достъп до други мрежови услуги.

2.4.Протоколни единици за данни





фигура 19 *Протоколни единици за данни*

На фигура 19 може да се види структурата на протоколната единица за данни и специалните и названия, спрямо нивата на OSI модела.

3. Протоколи

Адрес: **http://www.kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=16**

Протоколът е необходимо условие при осъществяването на комуникация. На това място се дава обща представа за характеристиките, основните функции, елементите и типовете съединения поддържани от протоколите. На фигура 18 е представен комуникационният сценарии на OSI модела, където се вижда, че обектите от един и същи слой на двата процеса комуникират на базата на протоколи. Това става възможно благодарение на съединения свързващи обекти от съседни слоеве участващи в действителния поток на конкретната комуникация. По тези съединения се предават масиви информация, наречени *протоколни единици за данни (PDU – Protocol Data Unit)*. Те са специфични за отделните слоеве. На всеки обект може да се предостави едновременно едно или няколко съединения с обекти от същия слой.

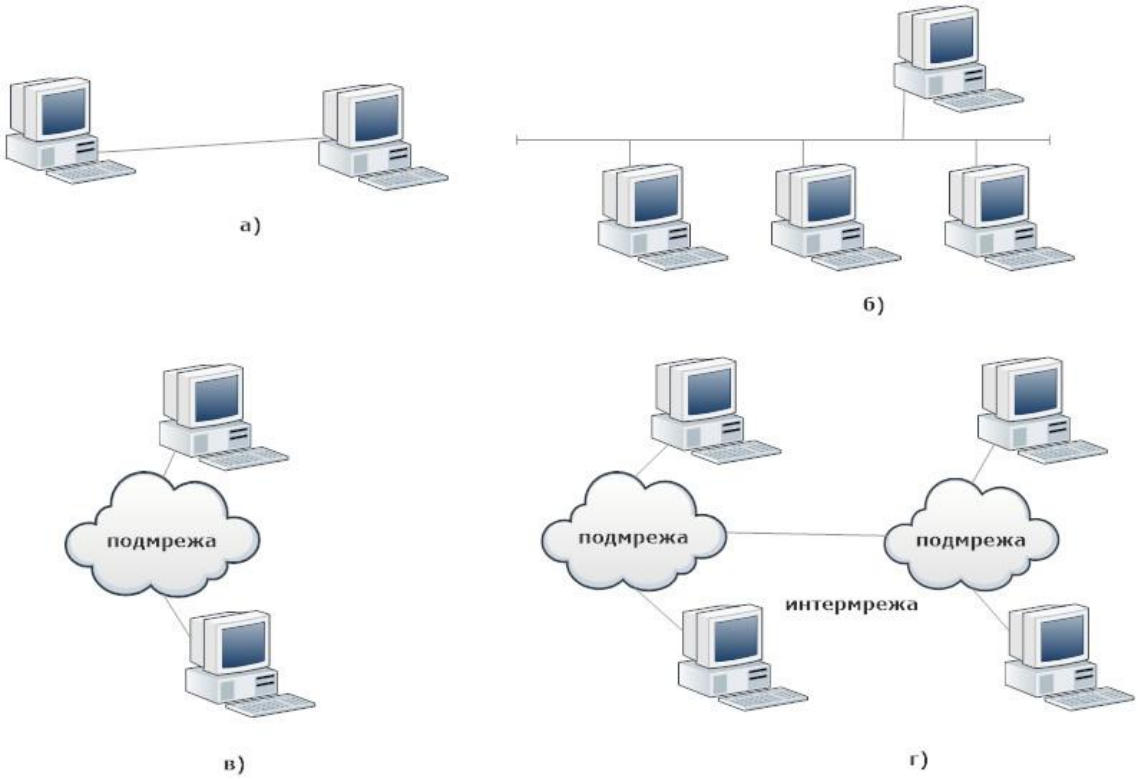
3.1.Съединения обслужвани от различните протоколи

Съществуват три типа съединения, обслужвани от протоколите:

- *симплекс (simplex)* - информацията по всяко време се предава само в едната посока;
- *полудуплекс (half duplex - HD)* – предаването в даден момент е само в едно направление, но и в двете посоки;
- *дуплекс (full duplex - FD)* – предава се в двете посоки едновременно.

Друга класификация (фигура 20) на съединенията е:

- точка – точка (например, при комуникация с модеми - фигура 20а);
- многоточково (LAN мрежи-фигура 20б);
- взаимодействие чрез подмрежа (фигура 20в);
- взаимодействие чрез интермрежа (например, Internet-фигура 20г).



фигура 20 *Съединения обслужвани от различните протоколи*

3.2.Елементи и характеристики на протоколите

Елементи на протоколите са:

- *синтаксис* – включва формата на данните, кодирането им и т.н;
- *семантика* – включва управляващата информация за координиране на обмена, допълнителна информация за контрол на грешките, възникнали при предаването им;
- *синхронизация* – включва изравняване на скоростта и определяне на последователността на предаване.

Характеристики на протоколите:

- *директност или индиректност* – директност при съединения от фигура 20а и фигура 20б, индиректност – фигура 20в и фигура 20г;
- *монолитност или структурираност* – монолитност е налице, ако комуникационните функции се изпълняват от един единствен протокол. Структурираността е налице, ако комуникационните функции се изпълняват от набор от протоколи, разпределени йерархично по слоеве (както при OSI модела);

- *симетричност или асиметричност* – симетричността е при локални мрежи с равноправен достъп. Асиметричност – при локални мрежи с отделен сървър;
- *стандартност или нестандартност* – стандартността е при тези протоколи, които са одобрени от международните стандартизиращи организации. Такива протоколи са протоколите на OSI модела.

3.3.Функции на протоколите

3.3.1. Фрагментация и дефрагментация

Обикновено трансферът на данни се състои в предаване на отделни последователни блокове с определена дължина. Протоколите трябва да разделят данните на блокове. Този процес е известен като *фрагментация (сегментация*). Блоковете се наричат PDU (Protocol Data Unit) – *протоколни единици за данни*. Всеки протокол има PDU със съответна дължина, като за някои е фиксирана, за други има максимално допустима стойност.

Примери:

- *стандарт X.25* – променлива дължина (максимум 4096 байта и 128 байта по подразбиране);
- *стандарт Frame Relay* – променлива дължина (максимум 1608 байта);
- *стандарт ATM* – фиксирана дължина 53 байта;
- *Ethernet* – променлива дължина (максимум 1526 байта).

В приемания край фрагментираните данни трябва да бъдат обединени за възстановяване на съобщението. Този процес се нарича *дефрагментация* (*десегментация)*.

3.3.2. Капсулация

Капсулацията е процес, при който протоколната единица на по-горен слой се вмъква в полето *<данни>* на протоколната единица на по-долен слой, след което към него се допълва служебна информация, необходима за извършване на предаването.

Служебната информация се дели на:

- *Адресна* - нужна за правилното доставяне на протоколната единица до крайния адресант;

- *Управляваща* - за правилното функциониране на протокола;
- *Контролна* - за откриване и/или коригиране на грешки, възникнали при предаването на протоколната единица.

3.3.3. Управление на съединението

Съществуват два режима на работа:

- без установяване на логическо съединение;
- с установяване на логическо съединение.

При логическото съединение има три фази:

- установяване на съединение;
- предаване на данни;
- разпадане на съединението.

3.3.4. Доставка на протоколни единици в правилен ред

Протоколните единици се номерират последователно и след получаването им се подреждат в правилния ред.

3.3.5. Управление на потока данни

Това е функция на приемника да ограничава големината и скоростта на протоколните единици, изпращани от предавателя. Често използван подход за тази функция е ARQ (Automatic Repeat reQuest) с неговите разновидности: „старт-стоп“ метод (Stop-and-wait ARQ). и метод на “плъзгащия се прозорец” (Go-Back-N ARQ). При първият от тях се изпраща пакет и се чака потвърждение за неговото получаване, след което се продължава с втория. При следващия метод се предават N кадъра един след друг (N-размер на прозореца), след което се чака потвърждение за тях.

Използват се за предаване на кадри и пакети на канално и транспортно ниво от OSI модела.

3.3.6. Контрол на грешките

Свързана е с откриването и/или коригирането на грешки при предаването на протоколните единици. Повечето протоколи използват само режими за откриване на грешки. След което очакват повторение на сгрешените протоколни единици. Този режим е по-прост и евтин за реализиране. Режимът за корекции на грешки се използва по-рядко и само,

ако не съществува възможност за обратна връзка (комуникационен канал), по-който приемникът да сигнализира за получени сгрешени протоколни единици.

Съществува и хибриден режим, при който се извършва частично коригиране с откриване на останалите грешки.

3.3.7. Адресация

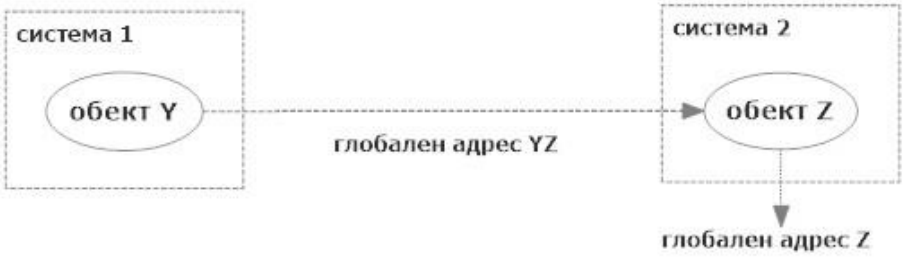
- *ниво на адресация* – всеки слой на комуникационния модел да има свой адрес т.е. имаме 7 нива за адресация на OSI модела.

Например, мрежовият слой използва *IP* адреси (мрежови адреси), каналния слой използва канални адреси (*MAC* адреси в локалните компютърни мрежи).

- *обхват* – два вида адреси: локални и глобални.

Например, *локални* са MAC адресите и X.25 адресите, *глобални* (уникални) са IP адресите.

- *идентификатори на съединението*-логическото съединение получава номер отличаващ го от другите (фигура 21).



фигура 21 *Идентификатор на съединение*

- *режим на адресация* – три типа адреси:

- *индивидуален адрес (unicast)* - съобщенията се предават само към един възел;
- *групов адрес (multicast)* - използва се за адресиране на група възли с цел предаване на едно съобщение на всички от групата едновременно;
- *общодостъпен адрес (broadcast)* - предаване до всички възли в мрежата или област.

3.3.8. Мултиплексиране

Това е процес на изграждане на много едновременни съединения в една система.

3.3.9. Услуги на предаването

Съществуват следните услуги:

- *Приоритетност* – по-високи приоритети се присвояват на тези съобщения, които трябва да бъдат предадени с минимално закъснение, напр. управляващите съобщения;
- *Праг на закъснението*– дефинира се за данни, чувствителни към закъснение;
- *Сигурност на данните*– за целта се използват методи като: шифриране на данните, ограничаване на достъпа до информацията с пароли и права на потребителите;
- *Максимална производителност* на процесорите на мрежовите възли.

4. Шумоустойчиво кодиране на цифрови съобщения

Откриването и коригирането на грешки в телекомуникациите намира приложение при запазване на правилността на предаваните данните по зашумени канали. За целта се използва т.нар. шумоустойчиво кодиране. То е един от трите вида кодиране на цифрови съобщения. Видовете кодиране са:

1. първично – обхваща началното представяне на съобщението в код. Например, превръщането му в ASCII код;
2. шумоустойчиво – към информационните битове на съобщението се добавят контролни битове за откриване и/или коригиране на грешки, възникнали при предаването му по комуникационния канал;
3. кодиране в комуникационната линия – кодира сигнала по начин, подходящ за предаване по физическата среда.

Основните изисквания към шумоустойчивите кодове са:

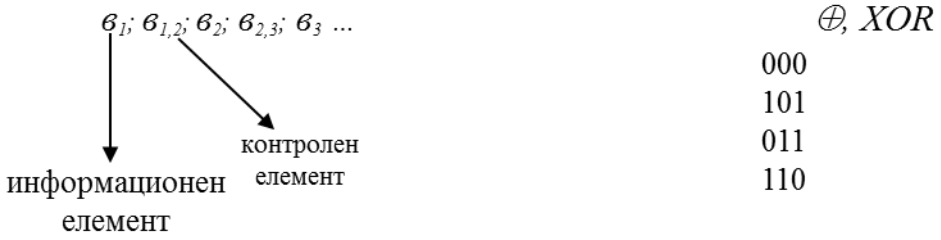
- да осигуряват висока информационна скорост (описана е по-долу);
- да реализират максимална коригираща способност при зададена сложност и бързодействие на декодера.

4.1. Класификация на шумоустойчивите кодове

Шумоустойчивите кодове се делят на две големи групи:

1. Непрекъснати кодове – не разделят предаваната информация, а вмъкват контролни елементи в определен ред между информационните. Делят се на:

- рекурентни кодове – в най-простия вариант информационният елемент се редува с контролен елемент.



фигура 22 *Рекурентни кодове*

На фигура 22 $в_i \in \{0,1\}$, а $в_{i,i+1}=в_i \oplus в_{i+1}$ е контролен елемент. В общия случай контролните елементи се формират чрез събиране по mod2 (XOR) на два информационни елемента, намиращи се на разстояние **i** един от друг, където **i** се нарича стъпка на събирането. Тя се определя на базата на статистическата информация за използвания канал. Зависи от "паметта" на канала.

При този код грешката в елемент b_i ще доведе до грешка в равенството за двата съседни контролни елемента $b_{i-1,i}$ и $b_{i,i+1}$. За правилното функциониране на кода е необходимо между два грешно приети елемента да има поне три вярно приети.

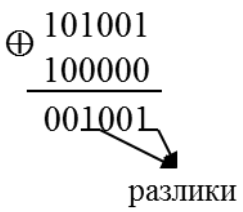
- конволюционни кодове (дървовидни) – при тях всеки k-битов информационен символ, постъпващ на входа на кодера (има k входа и n изхода), се трансформира в n-битов символ ($n \geq k$, $n=k+r$), където k/n е скоростта на кода (code rate), а r са контролните битове.

2. Блокови кодове – информационната поредица се разбива на отделни блокове , които се кодират и декодират независимо една от друга. Делят се на:

- разделими кодове – информационните и контролните елементи заемат едни и същи места във всички кодови комбинации. Обозначават се като (n, k) – кодове, където n – общият брой на елементите в блоковата комбинация, k – брой на информационните елементи, r=n-k – броят на контролните елементи в комбинацията. Подгрупи на разделимите кодове са линейните и нелинейните кодове. Линейните кодове се използват в практиката в широка група от приложения. Към тях се причисляват цикличните (CRC) кодове и каскадните кодове.
- неразделими кодове – отсъства деление на информационни и контролни елементи.

4.2.Основни понятия

Разстояние на Хеминг – броят на елементите, по които две кодови комбинации се различават. Използва се операцията сумиране по mod2 (XOR, символ на операцията - $\oplus$).



фигура 23

Кодово разстояние - минималното от всички разстояния на Хеминг за дадения код. Означение - d_0.

Тегло на кодовата комбинация – броят на единичните битове в нея.

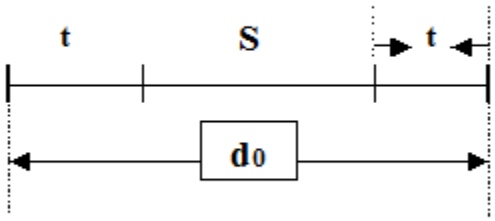
Вектор на грешката (e) – двоична кодова комбинация, съдържаща единични битове на позициите на грешките и нули, където няма грешки за предаваната кодова комбинация (фигура 23).

Спектър на кода – списък на разрешените кодови комбинации, разпределени по тегла, т.е. колко разрешени кодови комбинации на дадения код съответстват на всяко тегло.

Шумоустойчивите кодове се използват в следните режими [94]:

- **Режим на откриване на грешки –** най-често използваният режим. Използват се основно циклични и итерационни кодове, поради по-простата реализация на кодека. Код с кодово разстояние d_0, работещ в режим на откриване на грешки, може гарантирано да открива всякакви конфигурации от не повече от d_0-1 грешки във всяка своя кодова комбинация;
- **Режим на коригиране на грешки –** този режим се използва по-рядко и само ако обратната връзка е невъзможна или нецелесъобразна. В тези случаи се използват рекурентни, итерационни или каскадни кодове. Код с кодово разстояние d_0, работещ в режим на коригиране на грешки, може гарантирано да коригира до $\left[\frac{d_0-1}{2}\right]$ грешки във всяка своя кодова комбинация ([x] – цялата част на числото x).
- **Режим на частично коригиране с частично откриване на грешки** – при този режим код с кодово разстояние d_0 може да коригира t грешки включително ($0 \le t \le \left[\frac{d_0-1}{2}\right]$) и да открива конфигурации от s грешки ($t < s < d_0-t$) във всяка своя кодова

комбинация. Графичното представяне на този режим е показано на фигура 24.



фигура 24

Например, режимите на използване на един шумоустойчив код с кодово разстояние $d_0=5$ могат да бъдат:

- с откриване на всякакви конфигурации от не повече от 4 грешки;
- с коригиране на 1 или 2 грешки;
- с коригиране до 1 грешка и откриване на 2 и 3 грешки.

4.3.Линейни кодове

Линейните кодове образуват линейно пространство спрямо операцията сумиране по mod2. Това означава:

- затвореност спрямо операцията $\oplus$ - сумата по mod2 на кои да е две кодови комбинации дава валидна кодова комбинация;
- нулевата кодова комбинация е валидна кодова комбинация;
- d_0 е равно на минималното тегло от теглата на валидните ненулеви кодови комбинации.

Определения за най-често използваните в практиката двоични кодове:

- Двоична кодова комбинация (дума) с дължина **n** е последователността от **n** бита (0 и 1);
- Двоичен код **C** е множеството от кодови думи с дължина **n** бита;
- Векторът $\mathbf{c=(c_1c_2\ldots c_k\ldots c_n)}$, $\mathbf{c\in C}$ се нарича кодова дума (комбинация);
- Броят на информационните битове в кодираното съобщение е **k;**
- Броят на кодовите думи на **C** е $\mathbf{|C|=2^k;}$
- Параметърът **r=n-k** се нарича излишък на кода;

- Информационната скорост на кода **C** е величината 0<**R**<1, $\mathbf{R}=\frac{\boldsymbol{k}}{\boldsymbol{n}}$. Означението е (n,k) код.

Пример:

Нека е даден линейният код C={000,011,101,110}, за който n=3, |C|=4. Оттук следва, че $R = \frac{\log_2|C|}{n} = \frac{2}{3}$.

Кодът C е линеен $\Rightarrow \forall$x,y є C $\Rightarrow$ x$\oplus$y є C.

Този код не може да коригира грешки, понеже при възникване на единична грешка приетата двоична дума се различава в една позиция с повече от една валидна кодова дума.

$\hat{c} = (111)$ – сгрешена кодова комбинация $\Rightarrow d_0(\hat{c}, C)$={011,101,110}, което не може да определи точната вярна комбинация.

Пример:

Нека е даден линейният код C={000,111}, за който n=3, |C|=2. Оттук следва, че $R = \frac{\log_2|C|}{n} = \frac{1}{3}$.

Този код е пример за т.нар. код с трикратно повторение, при който е възможно коригирането на единична грешка.

c=(000) – предадена кодова комбинация, $\hat{c} = (001)$ – сгрешена кодова комбинация при приемане $\Rightarrow d_0(\hat{c}, C)$={000}, което означава коригиране до c=(000).

За кодиране на линеен код се използва образуваща матрица – $G_{k,n}$. За декодирането е необходима контролна матрица – $H_{r,n}$. Образуващата матрица в каноничен вид се записва по следния начин:

$$G_{k,n} = \|E_k|D_{k,r}\| = \begin{Vmatrix} 1 & 0 & \cdots & 0 & d_{11} & d_{12} & \cdots & d_{1r} \\ 0 & 1 & \cdots & 0 & d_{21} & d_{22} & \cdots & d_{2r} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & d_{k1} & d_{k2} & \cdots & d_{kr} \end{Vmatrix}$$, където E_k е единичната матрица с размерност **k** x **k**, а $D_{k,r}$ е допълнителна матрица, която може да се получи по произволен начин, като се спазват правилата:

- всеки ред трябва да съдържа поне d_0-1 единици;

- Хеминговото разстояние между всеки два нейни реда трябва да е поне d_0-2.

Кодираният вектор c=a.$G_{k,n}$, където a е вектора на съобщението.

Контролната матрица може да се получи от допълнителната матрица $D_{k,r}$ и единичната матрица E_r с размерност **r** x **r** по следния начин:

$$H_{r,n} = \left\| D_{k,r}^T | E_r \right\| = \left\| \begin{matrix} d_{11} & d_{21} & \cdots & d_{k1} & 1 & 0 & \cdots & 0 \\ d_{12} & d_{22} & \cdots & d_{k2} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ d_{1r} & d_{2r} & \cdots & d_{kr} & 0 & 0 & \cdots & 1 \end{matrix} \right\|$$

Основният метод за декодиране на линейни блокови кодове е свързан с изчисляването на синдрома на приетия вектор: $s = \hat{c}.H^T$. Ако полученият синдром е нулев т.е s=(000) то двоичната дума е приета правилно или е възникнала неоткриваема грешка. В противен случай, на всеки ненулев синдром еднозначно е съпоставен вектор на грешката, който се използва за нейната корекция. Осигуряване на такава възможност зависи от броя на контролните елементи. Например, за коригиране на еднократни грешки броят на синдромите трябва да бъде n+1 (n-за грешки във всяка позиция и един за вярно предаване). Тогава броят на контролните битове трябва да бъде $r \geq \log_2(n+1) \Rightarrow 2^r \geq (n+1)$, r=n-k $\Rightarrow 2^k \leq \frac{2^n}{n+1}$. Това неравенство ни дава възможност при зададен брой k на информационните битове в кодовата комбинация да подберем нужната й дължина n.

Следващият пример ще представи този подход на кодиране и декодиране.

Пример:

1. Да се генерира линеен код, съдържащ във всяка своя комбинация 5 информационни бита и коригиращ еднократна грешка.

От условието k=5, а $\frac{d_0-1}{2} = 1$ (условие за коригиране на еднократна грешка - t=1) $\Rightarrow d_0$=3. За осигуряването на достатъчно синдроми трябва да е изпълнено условието $r \geq \log_2(n+1) \Rightarrow 2^r \geq (n+1)$, r=n-k $\Rightarrow 2^k \leq \frac{2^n}{n+1}$. Понеже n>k то за целта k приема стойност 5, а n се замества последователно със стойности, започващи от 6, докато се удовлетвори неравенството. При n=9 условието е изпълнено, което означава, че кодът е (9,5), а r=9-5=4.

Според дефинираните по-горе правила за образуващата матрица $G_{5,9}$ придобива следния вид:

$$G_{5,9} = \|E_5|D_{5,4}\| = \begin{Vmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{Vmatrix}$$, където

условията за $D_{5,4}$ са:

- всеки ред съдържа поне d_0-1=3-1=2 единици;
- Хеминговото разстояние между всеки два нейни реда е поне d_0-2=3-2=1.

2. Да се кодира информационната поредица 11011с новополучения код (9,5) от предходния пример.

$a(a_1a_2a_3a_4a_5)$=(11011),

$$c(c_1c_2c_3c_4c_5c_6c_7c_8c_9)=a.G_{5,9}= (11011).\begin{Vmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{Vmatrix} =$$

=(110111101), където $c_i=a_1.g_{1,i}\oplus a_2.g_{2,i}\oplus a_3.g_{3,i}\oplus a_4.g_{4,i}\oplus a_5.g_{5,i}$

$\Rightarrow$ кодираният вектор е c=(110111101).

3. Да се приеме двоичната последователност от т.2.

За целта образуваме контролната матрица $H_{4,9}$, която придобива следния вид:

$$H_{4,9} = \|D_{5,4}^T|E_4\| = \begin{Vmatrix} 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{Vmatrix}$$

Синдромът получаваме от формулата:

$$s=H_{4,9}.c^T=\begin{Vmatrix}1&1&1&1&0&1&0&0&0\\1&1&1&0&1&0&1&0&0\\1&1&0&1&1&0&0&1&0\\1&0&1&1&1&0&0&0&1\end{Vmatrix}.\begin{Vmatrix}1\\1\\0\\1\\1\\1\\1\\0\\1\end{Vmatrix}=\begin{Vmatrix}0\\0\\0\\0\end{Vmatrix}\Rightarrow$$

синдромът е нулев, което означава, че комбинацията е приета правилно.

4. Да се допусне грешка в третия бит (отляво-надясно) т.е. приетият вектор да е $\hat{c}$=(111111101).

$$s=H_{4,9}.\hat{c}^T=\begin{Vmatrix}1&1&1&1&0&1&0&0&0\\1&1&1&0&1&0&1&0&0\\1&1&0&1&1&0&0&1&0\\1&0&1&1&1&0&0&0&1\end{Vmatrix}.\begin{Vmatrix}1\\1\\1\\1\\1\\1\\1\\0\\1\end{Vmatrix}=\begin{Vmatrix}1\\1\\0\\1\end{Vmatrix}\Rightarrow$$

синдромът не е нулев.

Съпоставяме го със стълбовете на контролната матрица $H_{4,9}$ за намиране на съвпадение. В този случай полученият синдром съвпада с третия стълб на матрицата, което означава, че трябва да се инвертира третият бит на $\hat{c}$. Новополученият вектор $\hat{c}'$=(110111101) е вярната последователност.

4.3.1. Код на Хеминг

Нека h е цяло число и h>=2. Двоичният систематичен линеен (n,k)-код с дължина на кодовата дума $n=2^h-1$ бита, дължина на информационния блок $k=2^h-1-h$ бита и минимално кодово разстояние $d_0=3$, имащ контролна матрица, на която колоните се състоят от всички ненулеви двоични думи с дължина h, се нарича код на Хеминг.

Пример:

Нека $h=3 \Rightarrow n=2^h-1=2^3-1=7$, $k=2^h-1-h=2^3-1-3=4 \Rightarrow$ двоичен линеен (7, 4) -код на Хеминг.

Контролната матрица $H_{r,n}$ съгласно определението ще бъде:

$$H_{3,7}=\begin{Vmatrix}1 & 1 & 1 & 0 & 1 & 0 & 0\\ 1 & 1 & 0 & 1 & 0 & 1 & 0\\ 1 & 0 & 1 & 1 & 0 & 0 & 1\end{Vmatrix}$$

На базата на тази матрица и дефинираните по-горе правила можем да получим образуващата матрица $G_{k,n}$:

$$G_{4,7}=\begin{Vmatrix}1 & 0 & 0 & 0 & 1 & 1 & 1\\ 0 & 1 & 0 & 0 & 1 & 1 & 0\\ 0 & 0 & 1 & 0 & 1 & 0 & 1\\ 0 & 0 & 0 & 1 & 0 & 1 & 1\end{Vmatrix}$$

4.3.2. Линеен код с една проверка по четност

Обозначава се като (n,n-1)-код (single parity check code). При него към всяка последователност от n-1 информационни бита се добавя един контролен бит, който е равен на сумата им по mod2 т.е.:

$c_1=a_1$, $c_2=a_2$, … $c_{n-1}=a_{n-1}$, $c_n=a_1 \oplus a_2 \oplus \ldots \oplus a_{n-1}$

Всяка кодова комбинация съдържа винаги четен брой единични елементи. Кодово разстояние е $d_0=2$ (едно за информационния бит и едно за контролния бит). Открива комбинация от нечетен брой грешки (еднократна, трикратна, петкратна и т.н.).

4.4. Циклични (Cyclic Redundancy Check - CRC) кодове

Намират широко приложение в практиката. Притежават следните основни свойства:

- ако кодовата комбинация c_1, c_2 … c_n, принадлежи на цикличен код, то и c_n, $c_1 \ldots c_{n-1}$ и т.н. също принадлежат на дадения цикличен код.
- съществува единствен нормиран полином $g(x) \in C$ от най-ниска степен, на който се делят без остатък всички кодови комбинации, разрешени за даден цикличен код. Забранените комбинации генерират остатък при делението. Всяка кодова комбинация може да се представи като полином, като всеки неин елемент се

разглежда като коефициент пред степен на променливата x, съответстваща на мястото на дадения елемент в комбинацията.

Пример:

c=(110111) $\Rightarrow$ $c(x)=1.x^5+1.x^4+0.x^3+1.x^2+1.x^1+1.x^0$ т.е. всеки полином може да се отъждестви с наредената (n+1)-орка ($c_0c_1\ldots c_n$).

Множеството от всички полиноми отговарящи на допустимите кодови комбинации на даден цикличен код образува поле на Галоа - GF(x), в което действията над коефициентите на полиномите се извършват по mod2.

Полиномът g(x) на (n,k) цикличен код C удовлетворява следните условия:

1. g(x) ∈ C е нормиран полином от най-ниска степен;
2. g(x) дели без остатък x^n-1;
3. степента му е равна на броя на контролните елементи r = n-k в кодовата комбинация;
4. осигурява максимален брой остатъци, за корекция на различните грешки;
5. ако $g(x)=g_0+g_1.x+\cdot\ \cdot+g_{r-1}x^{r-1}+x^r$, то образуващата матрица на този подклас линейни кодове се получава чрез циклично преместване на образуващия полином g(x):

$$G=\begin{Vmatrix} g_0 & g_1 & \cdots & g_{r-1} & 1 & 0 & \cdots & 0 \\ 0 & g_0 & \cdots & d_{r-2} & d_{r-1} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & g_0 & g_0 & \cdots & 1 \end{Vmatrix}$$

Алгоритъм за кодиране на информационна поредица чрез CRC код:

- полиномът a(x), съответстващ на информационната поредица, която трябва да се кодира, се умножава с x^{n-k}.
- полученият полином $a(x).x^{n-k}$ се дели на образуващия полином g(x) и се намира остатъкът от делението r(x).
- полиномът r(x) се прибавя (по mod2) към делимото $a(x).x^{n-k}$. $F(x)=a(x).x^{n-k}\oplus r(x)$ – задава двоичната последователност за предаване.

Пример:

1. Да се кодира информационната поредица 1011 чрез използване на цикличен (8,4) – код с образуващ полином – $g(x)=x^4+x+1$.

Стъпка 1: Образуващият полином се представя като двоична последователност от коефициентите пред степените на x: $g(x)=x^4+x+1 \Rightarrow g=(10011)$.

Стъпка 2: Информационната поредица 1011 се измества наляво с r=n-k=4 нулеви бита: 1011→10110000.

Стъпка 3: Получената двоична последователност се дели на g(x) и се определя остатъкът от делението r(x):

$$\begin{array}{r} 10110000 \\ \oplus\ 10011 \\ \hline 10100 \\ \oplus\ 10011 \\ \hline 1110 \end{array} \quad \begin{array}{|l} 10011 \\ \\ \\ \\ \\ \longrightarrow \text{остатък} \end{array}$$

Стъпка 4: Полученият остатък се събира по mod2 с делимото:

$$\begin{array}{r} 10110000 \\ \oplus\ \quad 1110 \\ \hline 10111110 \end{array}$$

Получената последователност 10111110 се предава по канала (фигура 25).

фигура 25 *Кодиране*

Алгоритъм за декодиране на получената двоична поредица чрез CRC код в режим на коригиране само на еднократни грешки:

При декодиране всяка пристигнала в приемника комбинация се дели на образуващия полином, при което се определя остатъка от делението: F`(x):g(x)→s(x), където s(x) е синдром на цикличния код. Ако получената

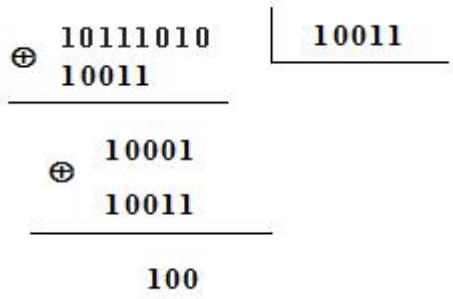
комбинация се дели без остатък на g(x) това означава, че тя принадлежи на множеството от разрешени кодови комбинации.

1. Определя се полиномът F`(x), съответстващ на приетата кодова комбинация;
2. F`(x) се разделя на образуващия полином g(x). Определя се остатъка от това деление – s(x).
3. Ако s(x)=0 то приетата кодова комбинация няма грешки. При s(x)≠0 се преминава към 4.
4. Определя се теглото wt(s(x)) т.е. на единичните елементи в комбинацията, съответстваща на s(x).
5. Ако wt е по-малко или равно от коригиращата способност на кода (в този случай t≤1) то коригирането се извършава чрез ⊕ на приетата кодова комбинация и остатъкът s(x) т.е. F`(x)⊕s(x). При t>1 се преминава към т.6.
6. Изпълнява се циклично преместване на една позиция на приетата кодова комбинация в избрана посока, след което новополученият полином отново се дели на g(x). Определя се теглото wt`(s`(x)).
7. Ако wt`≤1 се извършва операцията s`(x)⊕делимото, след което се извършва циклично преместване на резултата, в посока, обратна на предишните премествания. Крайният резултат представлява разрешена коригирана кодова комбинация. Ако wt`>1 се преминава към т.8.
8. Изпълнява се циклично преместване на делимото още една позиция в същата посока, след което се повтаря делението на g(x). Определя се wt` за новия остатък и се преминава към т. 7.

Пример:

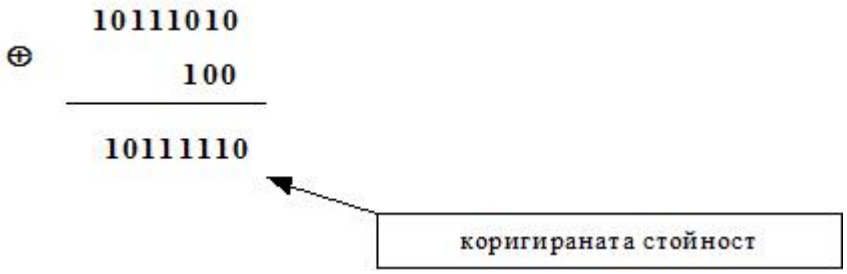
1. Да се декодира (с корекция за еднократна грешка) кодовата комбинация 10111010.

Стъпка 1: Приетата кодова комбинация F`(x): 10111010 се дели на образуващия полином g(x): 10011, s(x): 100.



Стъпка 2: Определя се теглото wt(100)=1, което отговаря на условието да не надвишава коригиращата способност на кода, в случая до една грешка.

Стъпка 3: Извършва се операцията F`(x)⊕s(x) и се получава коригираната стойност на получената кодова дума - 10111110:



5. Шифриране на данни

Предаването на данни по комуникационния канал изисква гарантирането на тяхната сигурност. Нейната цел е да реализира основните аспекти:

- поверителност (конфиденциалност) – защита на информацията от разкриване от неоторизирани потребители. Поддържат се различни степени на поверителност, в зависимост от важността на информацията;
- цялостност – гарантира, че данните не са променяни от неоторизирани лица;
- достъпност – осигурява достъп на оторизирани лица до данните и системните ресурси;
- автентичност – потвърждава, че опитът за достъп е от оторизираното лице, а спрямо информацията гарантира, че не е фалшифицирана.

Комуникационната мрежа и сигурността са обект на непрекъснати атаки. Могат да се разграничат [94] четири основни типа:

- прекъсване – свързана е с възпрепятстване на достъп на оторизирани лица до данните и системните ресурси. Този тип атака може да бъде реализирана физически или електронно;
- фабрикуване – неоторизиран обект се представя за действителен обект и генерира фалшива информация;
- прихващане – неоторизиран обект получава достъп до предаваната информация. Може да бъде реализирано чрез мрежово подслушване или послушване на радио сигнал;
- модифициране – неоторизиран обект променя информацията предавана от друг обект.

Прекъсването, фабрикуването и модифицирането могат да се причислят към активните атаки, защото за тяхната реализация се изискват активни действия от страна на атакуващия. Прихващането е пасивна атака, която трудно се открива.

За възпрепятстване на такъв тип атаки се използва шифриране на предаваните данни, при което се цели изходният им вид да се превърне в безсмислена поредица от символи, която да бъде предадена по

комуникационния канал. Тази процедура не гарантира отсъствието на атаки. Спрямо целевия обект и начина на използването му друга класификация ги определя като:

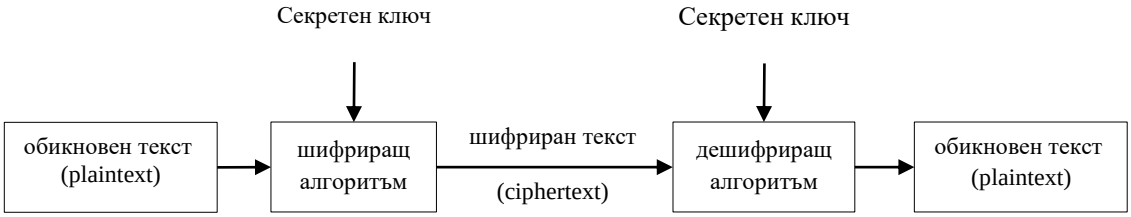
- атаки върху шифриран текст (ciphertext) – атакуващият е прехванал шифриран текст, който да анализира;
- атаки, използващи част от първичен текст на прехванато криптирано съобщение (known plaintext) – понякога атакуващият получава част от началния текст отговарящ на криптираното съобщение, което притежава. На базата на двойката (обикновен текст, шифриран текст) може да дешифрира криптираното съобщение. Някои алгоритми могат да бъдат добри срещу атаки върху шифрирания текст, но да не осигуряват сериозна защита при атаки, използващи двойката (обикновен текст, шифриран текст).
- атаки чрез подбран първичен текст (chosen plaintext) – тази ситуация възниква, когато атакуващият има възможност да се възползва от предлагана услуга за шифриране (използвана и от обекта-цел) да изпрати съобщение към себе си, включващо интересуващите го знаци с цел дешифриране на друго съобщение, от друг източник.

Алгоритмите за шифриране се разделят на три основни вида:

- симетрични алгоритми (Secret Key algorithms) [55][10][44];
- асиметрични алгоритми (Public Key algorithms) [73][78][9];
- хеш функции (Hash algorithms).

5.1.Симетрични алгоритми

Шифрирането, използващо симетрични алгоритми се нарича симетрично или конвенционално. Този тип шифриране използва един и същи секретен ключ от двете страни т.е за шифриране и дешифриране на съобщението (фигура 26).



фигура 26 *Симетрично шифриране*

Съществуват две основни изисквания към симетричното шифриране:

1. Двете страни да получат ключа по сигурен начин;
2. Да се запази секретността на ключа т.е. да остане известен само на двете страни. Това гарантира секретност дори ако е известен използвания шифриращия алгоритъм.

Освен горните изисквания Клод Шанън препоръчва алгоритмите да включват критерии като неяснота (confusion) и дифузия (diffusion). Неяснотата означава, че шифриращият алгоритъм скрива приетите модели на езика в изходния текст, а дифузията гарантира разбъркването на първоначалния текст, което би довело да объркване на атакуващия.

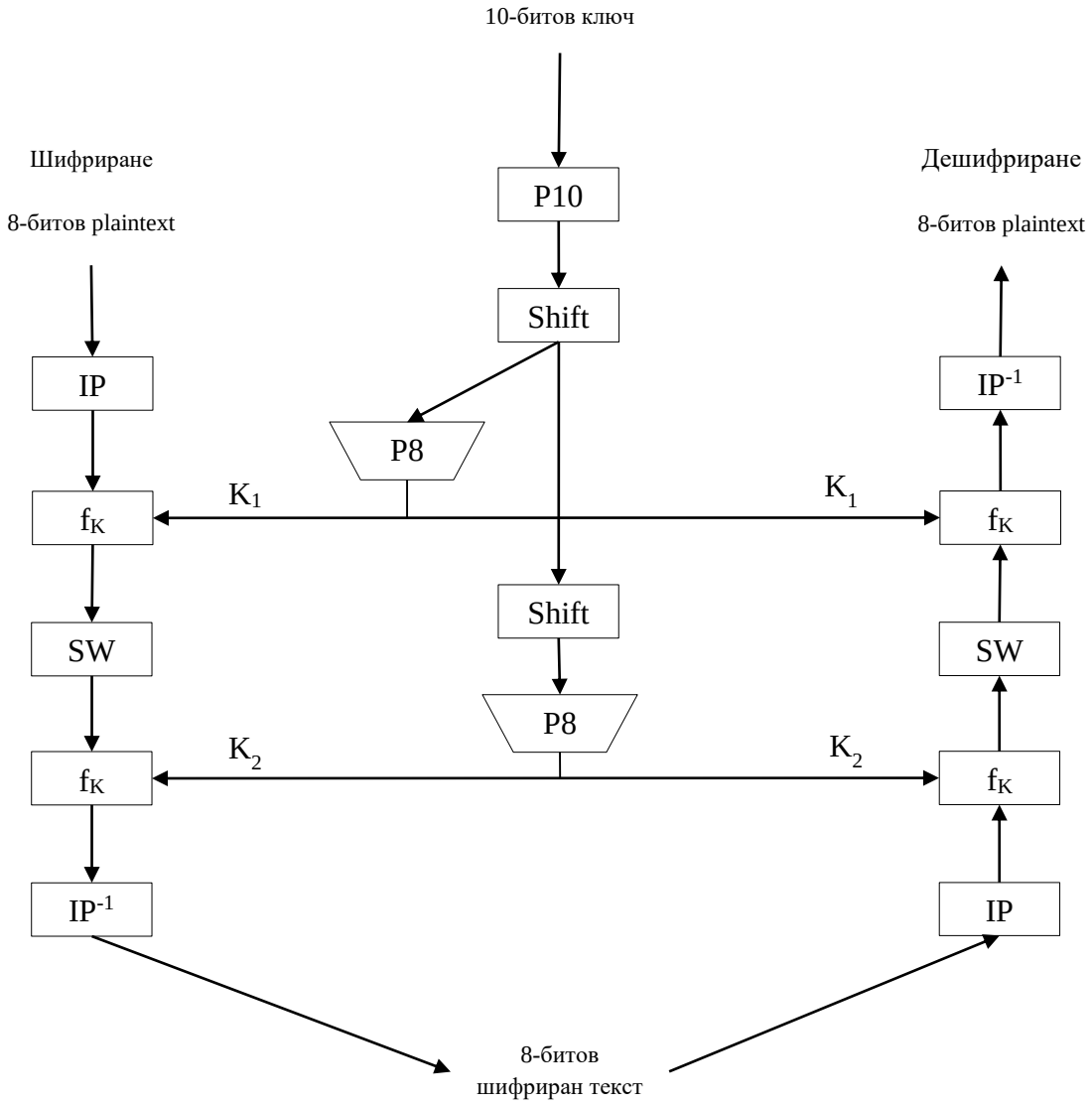
Най-често използваните подходи при реализирането на симетрично шифриране са прост алгоритъм с дълъг ключ или сложен алгоритъм с къс ключ.

Основните действия включени с процеса на шифриране са свързани с операциите изместване, пермутация и субституция. Пермутацията (разместване) е действие свързано с разместване на битове. Реализира се софтуерно или хардуерно чрез т.нар. P-кутии (Permutation box). Субституцията (заместване) замества една последователност от битове с друга. Това се осъществява чрез т.нар. S-кутии (Substitution box).

Пример за симетричен алгоритъм може да бъде Data Encryption Standard (DES) създаден от IBM и приет за използване през 1977 г. През 2001 г. е заменен от Advanced Encryption Standard (AES). За учебни цели се разглежда неговият опростен вариант Simplified Data Encryption Standard (S-DES).

5.1.1. Simplified Data Encryption Standard (S-DES)

На фигура 27 е представена схемата на функциониране на S-DES. Този алгоритъм е блоков, защото разделя данните на 8-битови блокове, които използва като входни данни заедно с 10-битовия ключ и генерира на изхода 8-битов шифриран текст. Този шифриран текст заедно със същия ключ се подава на входа на дешифратора, на чийто изход се получава първичния текст.



фигура 27 *Схема на функциониране на S-DES*

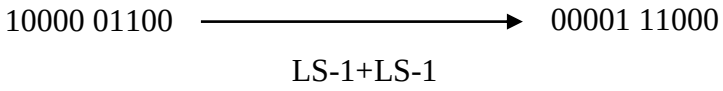
Шифриращият алгоритъм включва пет функции:

1. Инициализираща пермутация (IP) – свързана е с разместване на входните 8 бита от първичния текст. Например, могат да се разместят по схемата (2|6|3|1|4|8|5|7), където номерата показват оригиналните позиции

на входните битове. В т.5 върху тази новополучена подредба се прилага обратна пермутация, за да се получи ефекта $IP^{-1}(IP(x))=x$.

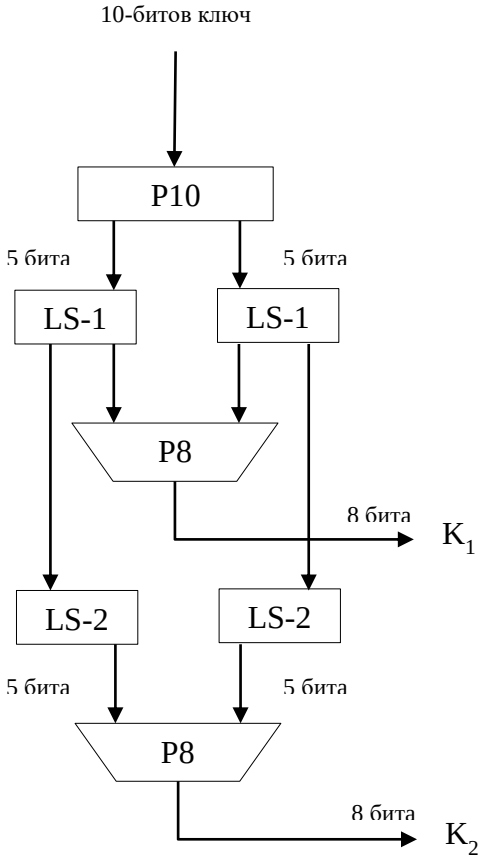
2. Сложна функция (f_K), включваща операциите пермутация и субституция и зависи от входния ключ.

Ключ: Тя използва два 8-битови подключа (фигура 29). За целта оригиналният 10-битов ключ се подлага на пермутация (P10). Например, ако ключът е последователността 1010000010 и се разместят позициите по схемата (3|5|2|7|4|10|1|9|8|6) то ще се получи вариантът 1000001100. Полученият резултат се разделя логически на две части по пет бита (10000 | 01100) и се извършва операцията изместване наляво с 1 бит (LS-1) на двете подчасти (фигура 28). Резултатът е (00001 | 11000).



фигура 28 *Изместване наляво с 1 бит (LS-1)*

За получаване на подключа K_1 се изпълнява пермутация P8 по следната схема (6|3|7|4|8|5|10|9). За примерът това означава K_1=10100100.



фигура 29 *Обработка на 10-битовия ключ*

Данни: 8-битовата последователност от данни (фигура 30) се разделя на леви 4 бита (L) и десни 4 бита (R). Нека F е функция, чийто резултат е 4-битов. Функцията f_K може да се зададе по следния начин:

$f_K(L,R)=(L\oplus F(R,SK),R)$, където SK е подключът K_1 или K_2.

При така зададената формула ако за вход на f_K зададем $f_K(L,R)$ ще получим следния резултат:

$f_K(f_K(L,R))=f_K(L\oplus F(R,SK),R)=(L\oplus F(R,SK)\oplus F(R,SK),R)=(L,R)$, защото $F(R,SK)\oplus F(R,SK)=0$

Функцията F изпълнява следните действия:

- входните 4 бита $b_1b_2b_3b_4$ се подлагат на операцията EP (expansion/permutation), която спазва схемата (4|1|2|3|2|3|4|1) относно позициите и връща 8-битов резултат $b_4b_1b_2b_3b_2b_3b_4b_1$;
- извършва се операцията $b_4b_1b_2b_3b_2b_3b_4b_1\oplus k_{11}k_{12}k_{13}k_{14}k_{15}k_{16}k_{17}k_{18}$, където $K_1=k_{11}k_{12}k_{13}k_{14}k_{15}k_{16}k_{17}k_{18}$ е 8-битовия подключ;

- ако полученият резултат се означи като ($r_{00}r_{01}r_{02}r_{03}r_{10}r_{11}r_{12}r_{13}$) то първите 4 бита се подават на S-кутия (S_0), а вторите 4 бита на S-кутия (S_1). Те спазват следните схеми спрямо позициите:

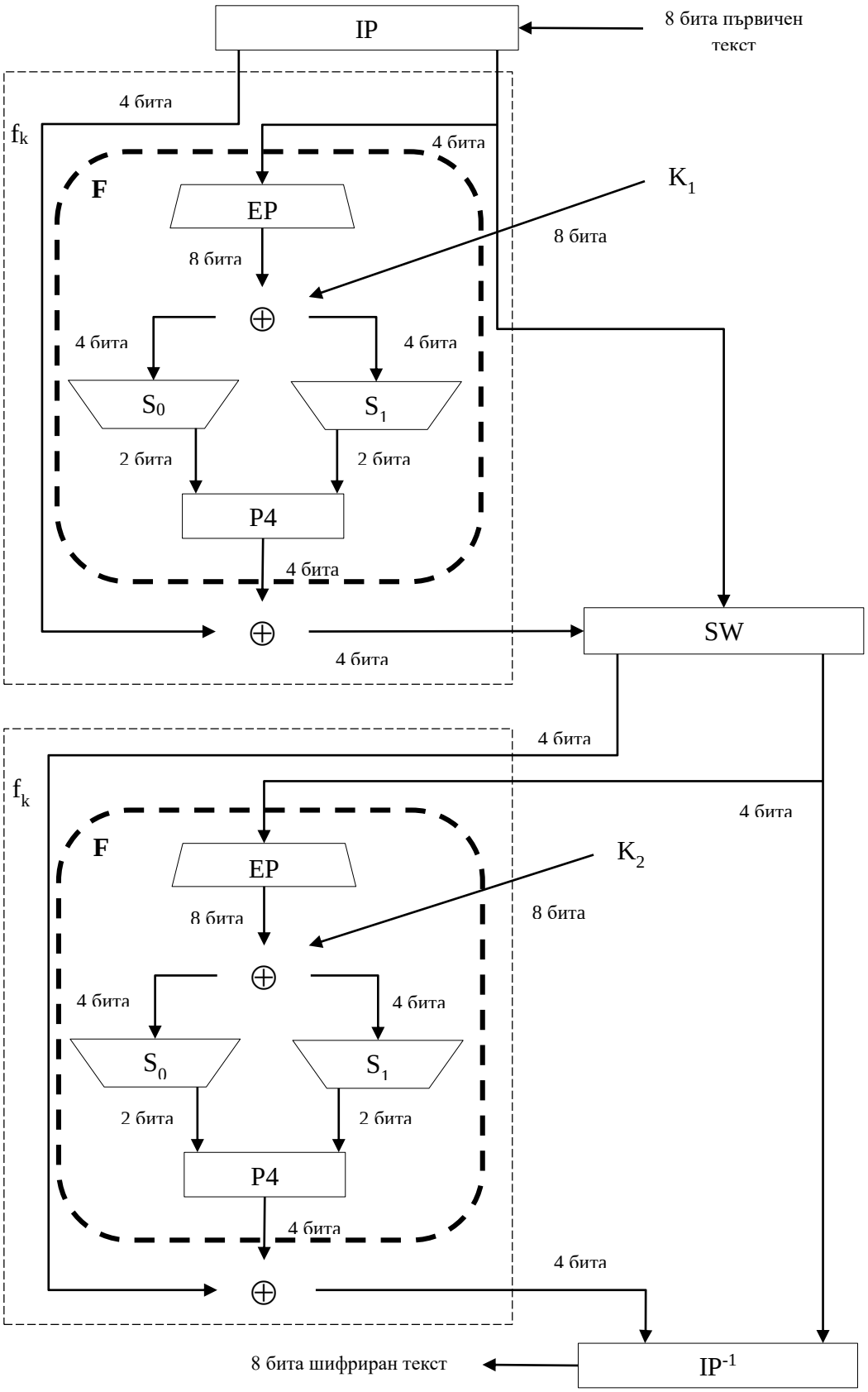
$$S_0 = \begin{bmatrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 2 \end{bmatrix} \text{ и } S_1 = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{bmatrix}$$

И двете S-кутии генерират по 2 бита резултат по следното правило:

- първи и четвърти входен бит определят 2-битов номер на ред т.е. за S_0 ако $r_{00}r_{03}$=00 то редът е с индекс 0;
- втори и трети входен бит указват 2-битов номер на колона т.е. за S_0 ако $r_{01}r_{02}$=10 то колоната е с индекс 2.

Тогава резултатът за S_0 ще бъде 3 или в двоичен вид 11.

След прилагане на същите действия за S_1 на изхода ще има 4-битов резултат върху, който се прилага пермутация P4 по схемата за позиции (2|4|3|1).

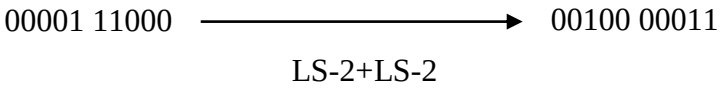


фигура 30 *Функционалности*

3. Проста пермутационна функция (SW), разменяща двете половини на данните. Това се налага от факта, че първата f_K променя само левите 4 бита от входните данни (8 бита), а по този начин следващото f_K (т.4) ще въздейства върху непроменените 4 бита (фигура 30).

4. Прилагане на f_K с нов подключ K_2.

Ключ: За конкретния пример изходният резултат за ключа от първата операция Shift се подава на втората операция Shift, която измества наляво с 2 бита (LS-2) двете му логически половини от 5 бита (фигура 31) и се получава последователността 0010000011. За получаване на подключа K_2 се изпълнява отново пермутация P8 по схемата (6|3|7|4|8|5|10|9) т.е K_2=01000011.



фигура 31 *Изместване наляво с 2 бита (LS-2)*

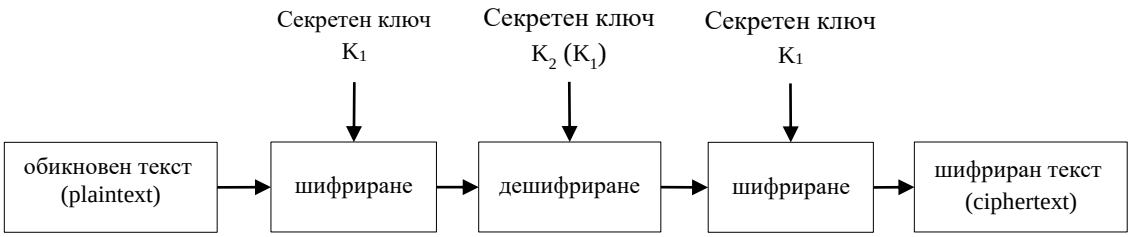
Данни: Функцията се прилага по начина описан в т.2 с нов подключ K_2 и входни данни от SW (т.3 и фигура 30).

5. Пермутационна функция, извършваща действие обратно на инициализиращата пермутация. Означена е като IP^{-1}. За приложената схема (2|6|3|1|4|8|5|7) от т.1 това означава разместване на позициите по следния начин (4|1|3|5|7|2|8|6).

Оригиналният DES използва 64-битов блок с входни данни и изпълнява 16 криптиращи операции. Ключът е 56-битов, от който се генерират 16 подключа от 48 байта.

5.1.2. Троен DES (triple DES)

Създаден е през 1979 г. за използване при шифриране на финансови операции. Използват се два ключа и трикратно използване на DES. Общата дължина на ключа е 112 бита. Шифрирането е в три фази показани на фигура 32.



фигура 32 *Троен DES*

Защо дешифриране?

Дешифрирането осигурява съвместимост със системи, използващи DES. Тогава в трите операции се използва един и същи ключ K_1, при което шифрираният текст на изхода се явява като резултат от единичен DES.

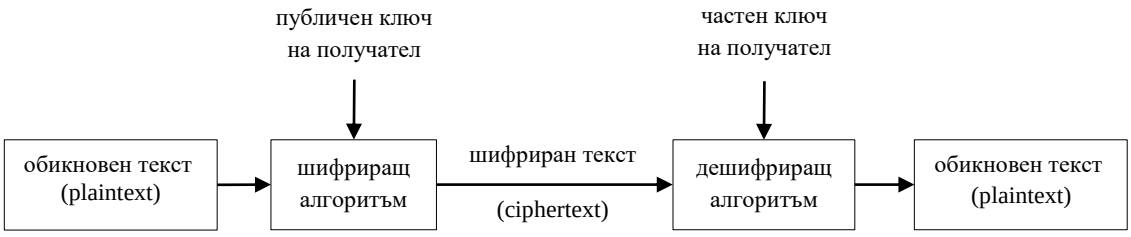
5.2.Асиметрично шифриране (шифриране с публичен ключ)

При симетричното шифриране има два основни проблема:

1. Разпространението на ключа и необходимостта от $\frac{n.(n-1)}{2}$ ключа при комуникация на n участника.
2. Реализацията на концепция за електронен подпис.

За решаването им се използва асиметричното шифриране, където се използват два различни ключа за шифриране и дешифриране. Единият ключ се нарича частен (напълно секретен), а другият публичен (известен на много други потребители). Съществуват два основни подхода:

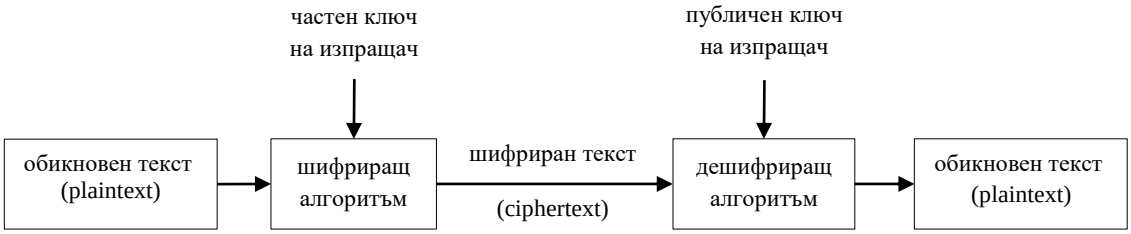
1. Шифриране – използва публичния ключ на получателя за криптиране на съобщение, предназначено за него. Получателят го дешифрира със своя частен ключ (фигура 33).



фигура 33 *Шифриране*

2. Автентикация (цифров подпис) – използва частния ключ на изпращача за шифриране на съобщение, предназначено за получател,

който притежава публичния ключ на изпращача за неговото разшифриране (фигура 34). Подходът се използва за потвърждаване на автентичността на съобщенията.



фигура 34 *Цифров подпис*

5.2.1. Алгоритъм RSA

RSA е един от първите шифриращи алгоритми, отговарящи на изискванията за асиметрично криптиране. Разработен е от Ron Rivest, Adi Shamir и Len Adelman (през 1977 г.), които получават за него награда Тюринг през 2003 г. Той е блоков алгоритъм с дължина на ключовете 1024 и 2048 бита. Първоначалното съобщение се разделя на k бита, където $2^k<n<2^{k+1}$. Шифрирането и дешифрирането могат да се изразят по следния начин: $C=M^e \bmod n$, $M=C^d \bmod n$, където M е блок от първичния текст, а C е блок от шифрирания текст. Двата обособени блока са числа принадлежащи на интервала [0,n-1].Наредената двойка (e,n) сформира публичния ключ на получателя, а (d,n) е частният ключ на получателя. Числата e, d, n се получават по следния начин:

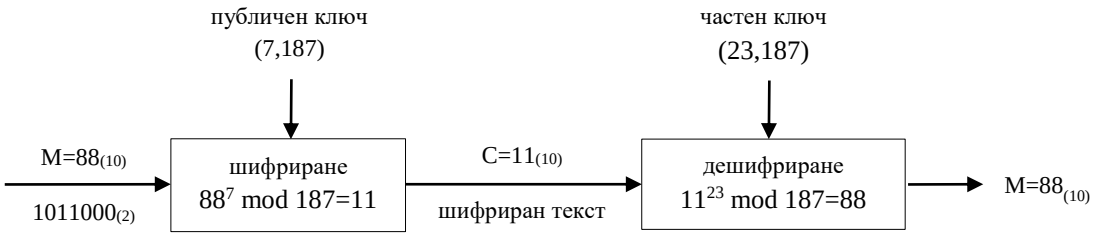
1. Генерират се две големи прости числа p и q (по-големи от 10^{100});
2. Образува се произведението n=p.q, което е част от генерираните ключове;
3. Изчислява се $\phi(n)=(p-1)(q-1)$, показващо броя на числата <n и взаимно прости с n;
4. Пресмята се e, за което $1<e<\phi(n)$ и $\gcd(\phi(n),e)=1$ (gcd – взаимно прости). То е част от публичния ключ;
5. Изчислява се $d=e^{-1} \bmod \phi(n)$ и $d<\phi(n)$, което е част от частния ключ;

RSA използва факта, че произведението n (от т.2) е почти невъзможно да се разложи на множители.

Функционирането на алгоритъма може да се демонстрира със следния пример: Да се изпрати съобщение M=88 като се използва RSA алгоритъм (фигура 35).

1. Нека p=17 и q=11;
2. n=p.q=17.11=187;
3. $\phi(n)=(p-1)(q-1)=16.10=160$;
4. Избира се e=7, така че да удовлетворява условията $1<e<160$ ($1<e<\phi(n)$) и $\gcd(160,7)=1$ ($\gcd(\phi(n),e)=1$);
5. Изчислява се $d=e^{-1} \bmod \phi(n)$ и $d<\phi(n)$ т.е. $d.e=1 \bmod 160$. За $d=23 \Rightarrow 23.7-160=1.160+1$ и $23<160$;
6. За шифриране на съобщението M=88 се използва публичният ключ $(e,n) \Rightarrow (7,187)$ и формулата $C=M^e \bmod n \Rightarrow C=88^7 \bmod 187=11$ т.е. шифрираното съобщение ще е 11;
7. За дешифриране на съобщението се използва частният ключ (d,n) $\Rightarrow (23,187)$ и формулата $M=C^d \bmod n \Rightarrow M=11^{23} \bmod 187=88$ т.е. дешифрираното съобщение ще е 88;

Забележка: В примера, за блока M, се използва директно цифровата стойност на съобщението в десетичен вид $88_{(10)}$. При по-подробно разписване това означава да се определи стойността на k (дължината на съобщението в битове), за която $2^k<n<2^{k+1}$. В случая при $n=187 \Rightarrow k=7$ ($2^7=128<187<2^{7+1}=256$). Тогава блокът M ще е съставен от 7 бита $\Rightarrow$ $88_{(10)}=1011000_{(2)}$ т.е. на входа на шифраторът ще е постъпила двоичната комбинация 1011000.



фигура 35 *Примерен RSA алгоритъм*

5.3.Хеш функции (Hash algorithms)

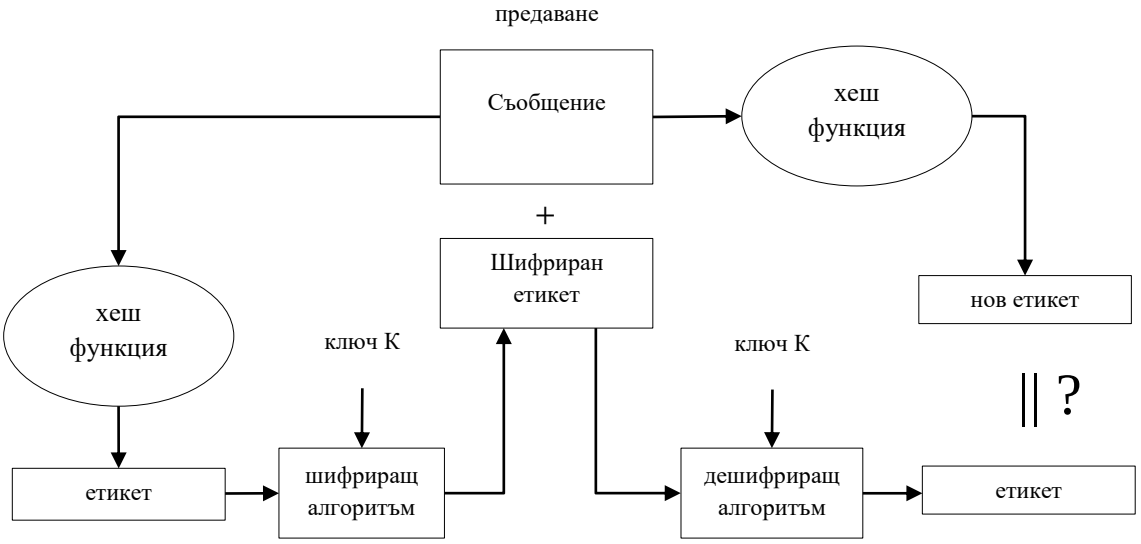
Хеш функциите са математически трансформации, които генерират число с фиксирана дължина от съобщение (представено в двоичен вид) с произволна дължина. Основни свойства на тези функции са:

1. За всяко двоично съобщение m лесно се изчислява резултатът h(m), което е практическо изискване;
2. При дадено h(m) не може да се определи m без да се пробва с всички възможни стойности на m и да се изчислят h(m). Това свойство се нарича еднопосочност (preimage resistant);
3. По зададено съобщение m трудно се намира m_1, за което h(m)=h(m_1). Свойството се нарича second preimage resistant;
4. Въпреки възможността различни стойности на m да се трансформират в едно и също h(m), на практика това е много трудно. Това свойство се нарича избягване на колизии (collision resistant).

Хеш функциите се използват за удостоверяване на източника (message authentication code - MAC) или удостоверяване на целостта на данните (modification detection code – MDC). При MAC се използва секретен ключ, а при MDC не се използва. Например, съществуват методи за потвърждаване на автентичността на съобщенията. Процесът включва проверка, гарантираща оригиналността на съдържанието на съобщението. Тук са представени симетрично и асиметрично шифриране на етикет.

5.3.1. Симетрично шифриране на етикет

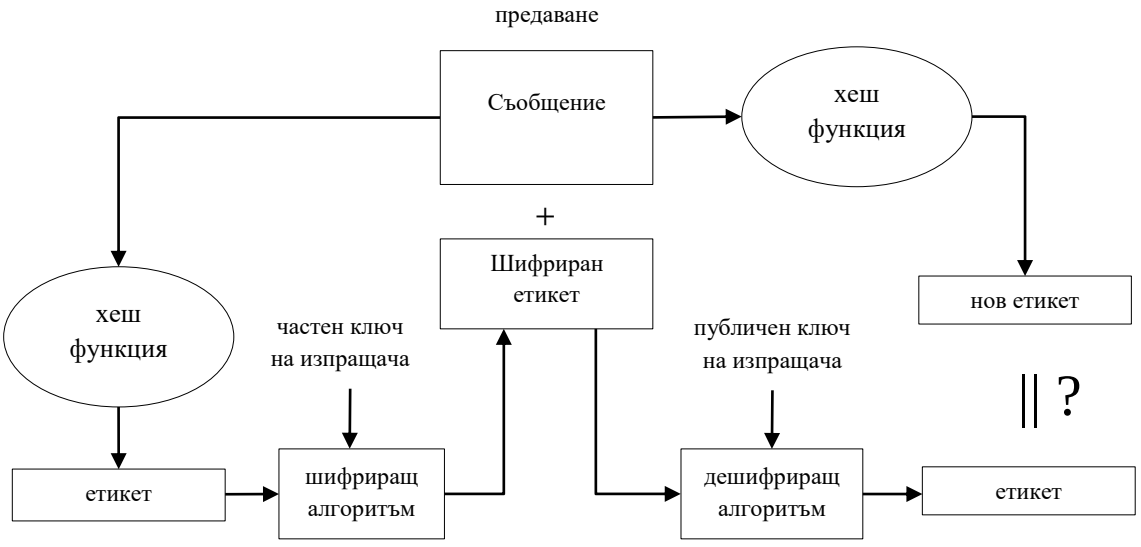
При този подход се използва ключът K за шифриране на етикета получен от хеш функция с входен параметър – съобщението. На входа на приемника шифрираният етикет се дешифрира със същия ключ и се сравнява с друг етикет, получен след използване на същата хеш функция с входен параметър - полученото съобщение (фигура 36). Ако двата етикета съвпаднат означава, че съобщението е оригинално.



фигура 36 *Симетрично шифриране на етикет*

5.3.2. Асиметрично шифриране на етикет

Подходът е подобен на горния с разликата, че имаме два ключа. Етикетът се шифрира с частния ключ на изпращача и се дешифрира с неговия публичен ключ (фигура 37).



фигура 37 *Асиметрично шифриране на етикет*

6. Локални компютърни мрежи

Мрежата е технология позволяваща на независими цифрови устройства да комуникират помежду си и да използват общи мрежови ресурси. В съвременното ежедневни е немислимо използването на компютъра като самостоятелна единица. Мрежата намира приложение навсякъде: в бизнеса, обучението и домакинството, затова целта на тема е да формулира някои основни понятия свързани с нея. Те трябва да гарантират еднозначно тълкуване на терминологията от новоизграждащите се специалисти. Локалната мрежа (LAN) гарантира пълноценното използване на локалните мрежовите ресурси. Умението за администрация и използване е абсолютно задължително за съвременните специалисти.

6.1.Определения

LAN е технология позволяваща на независими цифрови устройства да комуникират помежду си и да използват общи мрежови ресурси като процесорно време, памет, файлове и входно-изходни устройства.

Към основните характеристики на LAN могат да се причислят възможност за групово (multicast) и общодостъпно (broadcast) предаване, малка вероятност за грешки при предаването, стабилност при високо натоварване.

Според функциите, които изпълнява, краен възел на една локална мрежа може да бъде:

- Работна станция - произволен компютър или терминал, чрез който се осъществява достъп до мрежовите ресурси;
- Сървър - компютър, осигуряващ мрежовите ресурси, както и тези които обслужват мрежата.

Съществуват и междинни мрежови възли – повторители, концентратори, комутатори, мостове, маршрутизатори.

Един компютър може да изпълнява едновременно функциите на сървър и работна станция (равноправен достъп „peer-to-peer“). При мрежи с усилено използване на даден ресурс е желателно компютърът, предоставящ този ресурс, да се използва единствено и само като сървър.

Под „сървър“ се разбира приложен процес (програма), реализиращ дадена мрежова услуга. Трябва да се отбележи, че е възможно един компютър да обслужва няколко сървъра едновременно, стига да не се натоварва прекалено системата.

Видове сървъри:

- *Файлов* – програма, позволяваща достъп до файловата система на компютъра за съхранение и извличане на файлове и програми;
- *Сървъри за печат* – програма, осигуряваща достъп до принтера на компютъра, на който е стартирана;
- *Сървър за асинхронни комуникации* – програма стартирана на компютър с няколко модема, позволяващи използването им от всички потребители;
- *Сървър за отдалечен достъп (RAS)* – програма, стартирана на компютър, осигуряваща достъп до локалната мрежа от отдалечен компютър чрез модем и телефонна линия;
- *Сървър за електронна поща* – програма, управляваща електронните пощенски кутии на потребителите;
- *Факс-сървър* – програма, управляваща изпращането и получаването на факс съобщения;
- *Сървър за бази данни* – програма, реализираща функциите на ядрото на системата за управление на БД;
- *Сървър за приложни програми* – подава копие на приложната програма до работната станция по мрежата.

6.2.Основни предимства на LAN:

- Осигурява общи ресурси – намаляване на разходите за скъп хардуер;
- Повишаване на ефективността на сравнително непроизводителни компютри (пример – повечето дискова памет от сървъра);
- Възможност за използване на обща база данни;
- По-удобна колективна работа при разработването на проекти в група;
- Възможност за електронна комуникация (да комуникират без да напускат работното си място);
- Възможност за свързване към други LAN/WAN.

6.3.Физически слой в LAN

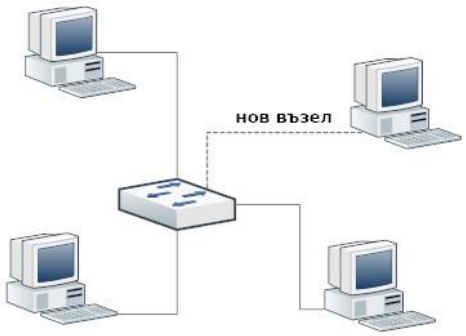
Мрежовият адаптер е хардуерната част при LAN изпълняващ функциите определени от първо и второ ниво на OSI модела. Към физическия слой могат да се причислят спецификациите на топологията, кабелната система и донякъде безжичната преносна среда.

Когато се ангажира терминът топология от съществено значение е да се уточни дали е физическа или логическа. Физическата топология (physical topology) указва физическото разположение на възлите и кабелната система, докато логическата топология (logical topology) се определя от начина на предаване на данните между устройствата, независимо от тяхното разположение. Двете топологии могат да са едни и същи или да се различават при една реализация на мрежата.

Като най-популярни физически топологии в LAN могат да се представят: звезда, дърво, пасивна шина, активна шина, кръг, решетка и хибридна топология.

6.4.Популярни видове LAN топологии

- звезда (star) – при нея крайните мрежови възли (работни станции, сървъри) са свързани към централен възел (комутатор) във вид на звезда (фигура 38).

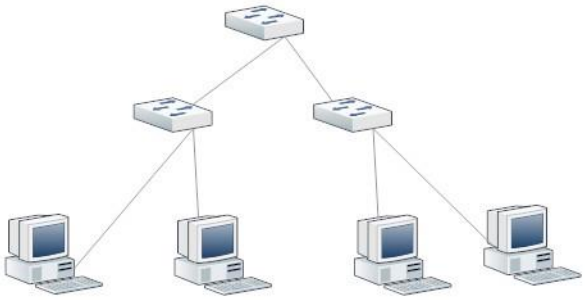


фигура 38 *Тип звезда*

Предимство: лесно добавяне на нов възел.

Недостатък: повреда в централния възел предизвиква разпад на мрежата.

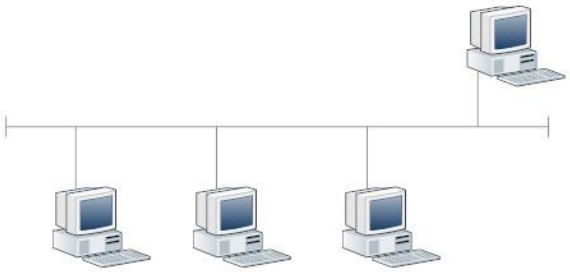
- дърво – дърво с корен (главен концентратор), към който са свързани крайни възли и концентратори, разположени на различни нива (фигура 39).



фигура 39 *Тип дърво*

Предимство: при повреда в концентратор от второ или по-долно ниво пропада само определен участък от мрежата.

- пасивна шина – една шина за данни (кабел), към която са свързани отделните мрежови възли (фигура 40).



фигура 40 *Тип шина*

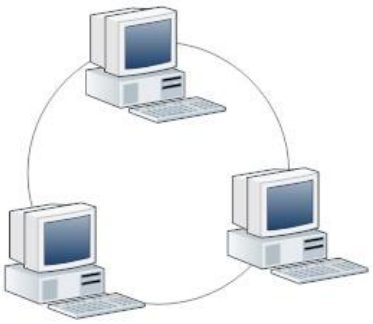
Предимства: лесно добавяне на нови възли; повреда на един възел не указва влияние върху другите.

Недостатъци: ограничено покритие; слаба диагностика на мрежата; прекъсване на шината води до разпадане на мрежата.

- активна шина – изходът на всеки възел към входа на следващия. За комутиране в две различни посоки са необходими две активни шини. Всеки възел действа, като регенератор и усилвател (фигура 41).

фигура 41 *Тип активна шина*

- кръг (ring) – възлите в мрежата са свързани в кръг (фигура 42).

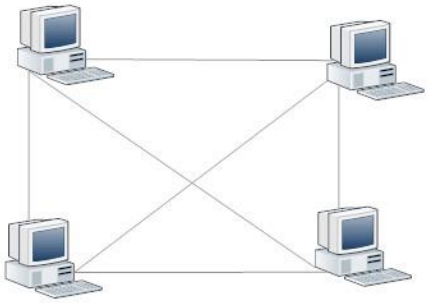


фигура 42 *Тип кръг*

Предимства: възможност за покриване на големи разстояния; удобство при използване на оптични влакна.

Недостатък: трудно се добавят нови възли.

- решетка (mesh) – всеки компютър е свързан с всеки друг (фигура 43).

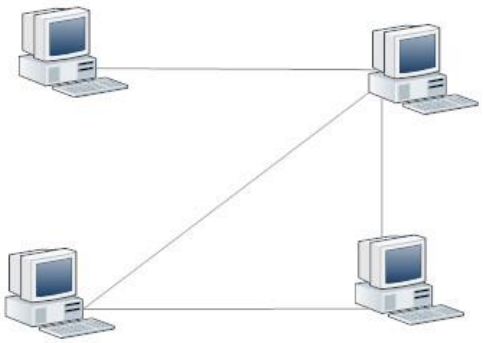


фигура 43 *Тип решетка*

Предимство: висока отказоустойчивост.

Недостатъци: висока цена за построяването ѝ; сложна реализация при повече компютри.

- Хибридна топология – базира се на полурешетъчна топология, където допълнителни връзки има само между някои от компютрите (тези, които се нуждаят най-много от отказоустойчивост на връзката)(фигура 44).

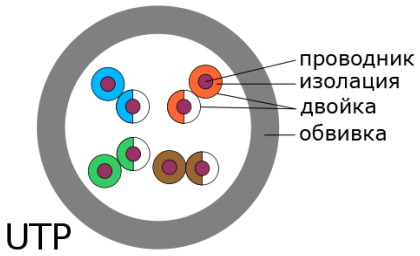


фигура 44 *Хибридна решетка*

6.5.Физически среди за разпространение на сигнала

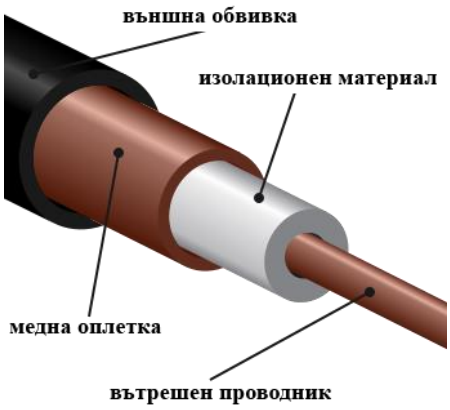
Комуникационният сигнал може да използва кабелна или безжична среда за своя пренос. Кабелната среда включва: коаксиален кабел, кабел с усукани двойки проводници и кабел с оптични влакна.

- *кабели с усукана двойка проводници* – състоят се от няколко (обикновено 4) двойки медни проводници. Двата проводника на всяка двойка са изолирани и взаимно усукани за намаляване на външните шумове (фигура 45). Предназначени са за малки разстояния. Съществуват различни категории, поддържащи различни скорости. Самата категория се идентифицира със символите Cat x и означава номера на успешно преминалия тест за производителност. В момента едни от най – разпространените категории са Cat 5e и Cat 6. Максималната дължина за сигнала без повторител е 100 м.



фигура 45 *Структура на UTP кабел*

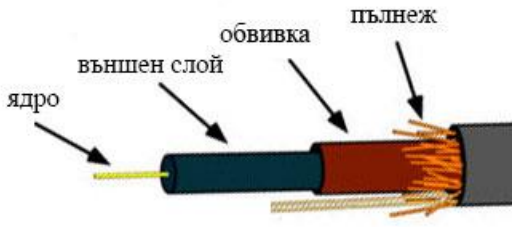
- *коаксиални кабели* – вътрешен проводник обвит с изолационен материал, медна оплетка и външна обвивка (фигура 46). Оплетката играе роля на предпазен екран. Ако съществува първи изолационен слой от фолио и втори от метална оплетка, то той е двойно екраниран. Покриват се по-големи разстояния и се поддържат по-големи скорости, защото имат по-добра защита от електромагнитни смущения.



фигура 46 *Структура на коаксиален кабел*

Съществуват различни типове и категории коаксиални кабели, предлагащи се от различните производители. Много от тях се използват от мрежи със специално предназначение. В ранните реализации на Ethernet, коаксиалният кабел беше най-популярния тип. Неговите два основни варианта са тънък (thinnet) и дебел (thicknet), присъщи съответно за топологиите 10Base2 и 10Base5.

- *Влакнестооптични кабели* – съставени са от отделни влакна направени от стъкло или пластмаса. Оптичното влакно се състои от ядро (core) и външен слой (cladding) с различен показател на пречупване на светлината (фигура 47). Това важно свойство на оптичното влакно не позволява преплитането на информация между отделните влакна в един кабел и позволява кабелът да се извива и усуква. Сигналите се пренасят по сърцевината под формата на модулирани светлинни импулси с дължина на вълната от инфрачервената област и не се влияят от външни електромагнитни смущения. Обикновено влакната се използват по двойки, като всяко носи сигнал само в едната посока.



фигура 47 *Структура на оптичен кабел с едно влакно*

Влакната, използвани в телекомуникацията, са най–често с диаметър 125 μm. Оптичният пренос, който осигуряват може да бъде многомодов (multimode) или едномодов (single mode). Ядрото на едномодовите влакна е

с диаметър 9 μm, докато при многомодовите е с диаметър 50 μm или 62,5 μm. Описват се с двойка числа, показващи диаметъра на влакното и на неговата обвивка (например, 62.5/125 микрона). Кабелът с многомодови влакна може да покрива разстояния до 300-500 метра, а с едномодови влакна до 80-200 км. При използване на една дължина на вълната за нишка, едномодовите оптични влакна достигат скорости от порядъка на 40 Gbps.

Безжичната среда позволява протичането на комуникацията да се извърши без наличието на кабел. Съществуващи технологии за пренос са:

- с широк радиоспектър – дава възможност честотната лента да се разделя на подканали, които поддържат самостоятелен пренос. Използвани техники са:

- *скачаща честота* (FHSS - Frequency-hopping spread spectrum) – разделя честотната лента на подканали. В даден момент използва само един канал. Сигналът скача според предварително уговорен ред и честота;
- *директна поредица* (DSSS - Direct-sequence spread spectrum)– разделя честотната лента на подканали. Използва различните подканали в пореден ред.
- *ортогонална честота* (OFDM - Orthogonal frequency-division multiplexing) – радиосигнала се разделя на множество подсигнали и едновременно се излъчва на съвсем леко различаваща се честота в общия канал.

- с тесен или еднолентов радиоспектър - използва само един канал за предаване.
- инфрачервени – поддържат два варианта на разпространение:

- директен - разпространява се в една посока;
- дифузен - разпръсква във всички посоки.

- лазерни – използват концентриран лъч, запазващ фокуса на далечно разстояние.

6.6.LAN стандарти

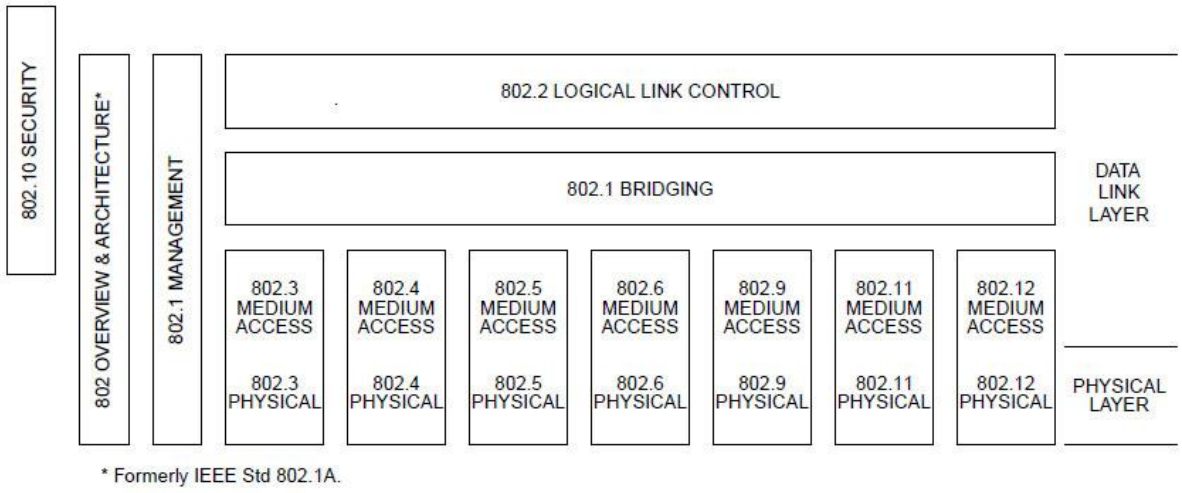
Съвместимостта между производителите на мрежови компоненти се гарантира от стандарти дефинирани от стандартизиращи организации. Едни от най-престижните са: ANSI (American National Standards Institute),

IEEE (Institute of Electrical and Electronic Engineers), ISO (International Organization for Standardization), IEC (International Electrotechnical Commission), IAB (Internet Architecture Board). IEEE е отговорен за дефинирането и публикуването на стандарти за телекомуникация и обмен на данни. Техен принос са стандартите за локални и градски мрежи (LAN и MAN), които обикновено се цитират като 802 серия (80-година 1980, 2-месец февруари) и се разработват от комитет с наименование Проект 802. Целта е улесняване на комуникацията между различните типове LAN и отделяне на физическата среда от протоколите. Това дава възможност различните LAN архитектури да се реализират върху различна кабелна среда. Стандартите от Проект 802 са организирани в трислойна йерархия, отговаряща на физическия и каналния слой на OSI модела. Каналното ниво на IEEE 802 се разделя на две поднива LLC и MAC. Това деление позволява да се разграничи достъпа до средата, реализиран от MAC подслоя, от управлението на потока данни, като позволява взаимно функциониране на 802 съвместими мрежи. Отделни екипи на стандартизиращата организация разработили стандарти в следните области:

- **802.1** – дефинира LAN и MAN съвместимост, мостове действащи в MAC подслоя и алгоритъм STA (Spanning-Tree algorithm) за предотвратяване на междумостово зацикляне;
- **802.2** – дефинира Logical Link Control (LLC) – спецификации за осигуряване на интерфейс между MAC подслоя и мрежовия слой;
- **802.3** – CSMA/CD – специфицира начина на функциониране на Ethernet мрежите, базирани на метода за множествен достъп с разпознаване на носещата и откриване на колизии;
- **802.4** – Token Bus – задава стандарт за мрежа използваща 75 омов коаксиален или оптичен кабел с предаване на маркер;
- **802.5** – Token Ring – задава стандарт за физическа топология звезда и логическа кръг, използваща усукана двойка проводници и метод на достъп с предаване на маркер;
- **802.6** – MAN – задава стандарт за регионална мрежа (Distributed Queue Dual Bus Access Method);
- **802.7** – Broadband – правила за изграждане на мрежи използващи технологии за широколентово предаване (FDM);

- **802.8** – Fiber Optics – задава стандарт за мрежи, използващи оптичен кабел за преносна среда;
- **802.9** – установява стандарт за предаване на глас и данни по ISDN;
- **802.10** – обхваща изграждането на частни виртуални мрежи (VPN);
- **802.11** – специфицира Wireless LAN Medium Access Control (MAC) и Physical Layer (PHY) за безжична комуникация;
- **802.12** – 100 VG AnyLAN – метод за достъп с приоритет на заявката (Demand Priority Access Method), разработен от HP с цел комбиниране на предимствата на Ethernet, Token Ring и ATM.

Някои от тези стандарти следва да бъдат разгледани по-подробно. Зависимостта между тях е онагледена чрез фигура 48, публикувана в изданието от 1998 година за IEEE 802.2.



фигура 48 *Зависимост между стандартите от Проект 2*

6.7.Канален слой в LAN по Проект 802

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=17**

За преодоляване на различията в мрежовите архитектури за LAN от серията IEEE 802.x е избран подходът с разделянето на каналния слой на двата подслоя:

- Горен подслой за управление на логическите канали (LLC – Logic Link Control);
- Долен подслой за управление на достъпа до комуникационната среда (MAC – Media Access Control).

Функциите на каналния слой се осъществяват от мрежовия адаптер.

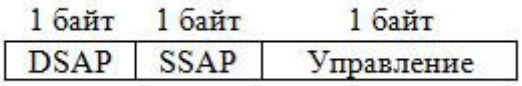
6.7.1. LLC подслой

LLC получава пакети от мрежовия слой, формира и предава номерирани LLC блокове, контролира грешките и допълнително може да определя последователността на кадрите и да коригира грешки получени при предаването. LLC подслоят е независим от MAC-подслоя и от комуникационната среда. Функционирането на LLC подслоя е описано в стандарта IEEE 802.2. Целта на LLC подслоя е да обменя информация между крайните потребители през локална мрежа, която използва 802-базирани MAC протоколи. LLC подслоя предоставя идентификация на ULP (upper-layer protocol), DLC (data link control) функции и услуги по връзката. Той е независим от топологията, средата на предаване и техниките за контрол на достъпа до преносната среда. LLC подслоят предоставя следните три услуги по комуникацията между SAP (service access points):

- Услуги без потвърждения (LLC1) (*дейтаграмен режим без потвърждение*) –мрежовите обекти могат да обменят PDU (protocol data units) без да осъществяват връзка на канално ниво (Data Link Layer - DLL). Преносът на данни може да бъде тип: точка към точка, точка много точки или разпръскване (broadcast). Надеждността на предаването се проследява от протоколите от по-горните слоеве. Например, TCP протоколът контролира потокът от данни и гарантира надеждността при предаване, затова необходимостта от подобни функции на канално ниво отпада. По този начин се избягват усложнения, свързани с протичането на комуникацията.
- Услуги с потвърждения от високо ниво (LLC2) (*режим на виртуално съединение*) – този набор от услуги предоставя средствата за създаване, използване, нулиране (рестартиране) и прекъсване на DLL връзки. Тази услуга предоставя също DLL поредни номера, контрол на потока и възстановяване от грешки, за надежден обмен на PDU единиците на установената връзка. Връзките са тип точка към точка. Тази услуга е полезна, когато в комуникацията участват прости устройства, които не поддържат протоколи от трето и четвърто ниво.
- Услуги с потвърждения от ниско ниво (LLC3) (дейтаграмен режим с потвърждение) – тези услуги предоставят средствата, чрез които единиците на мрежовия слой могат да обменят надеждно PDU, но без

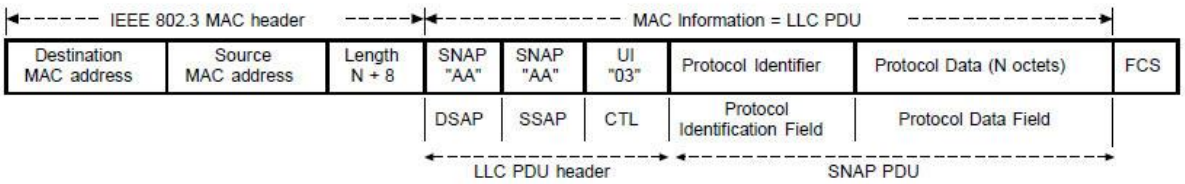
установяване на DLL връзка. Трансфера на информационни единици е от тип точка към точка.

Тези услуги се прилагат за комуникация между LLC подслоеве и са прозрачни за потребителите. На фигура 49 е показан хедъра на LLC блок, където DSAP (Destination service access point) и SSAP (Source service access point) идентифицират протокола от мрежово ниво и гарантират възможността да бъдат използвани различни протоколни стекове в една и съща мрежа (мултиплексиране на протоколите). Протоколите имат назначени шестнадесетични стойности, които се присвояват на тези полета.



фигура 49. *LLC хедър*

Полето за управление определя избора на механизми за адресиране и контрол на потока данни, които да се използват от участващите устройства в комуникацията, благодарение на изброените преди това три типа LLC услуги. За съвместимост с протоколи от по-горни нива е създадена подрамкова структура, наречена SNAP (Subnetwork Access Protocol) подрамка, която се състои от петбайтово поле. То се разделя на трибайтово поле наречено OUI (Organizationally Unique Identifier) и двубайтово поле (Type) за идентификация на протокола. Тази подрамкова структура следва LLC хедъра, който е трибайтов (фигура 49). Например, разположението спрямо останалите полета във формата на IEEE 802.3 MAC кадър е показано на фигура 50, означен като SNAP PDU.



фигура 50 SNAP PDU в IEEE 802.3 MAC кадър

Destination MAC address – шестбайтов MAC адрес на получателя
Source MAC address – шестбайтов MAC адрес на източника
Length – двубайтово поле за дължина. В нея се включва полето за данни (Protocol data – N байта), LLC хедъра (3 байта) и SNAP хедъра (5 байта)

DSAP – еднобайтово поле за идентификация на LLC точката за достъп до услугата получател
SSAP - еднобайтово поле за идентификация на LLC точката за достъп до услугата източник
CTL – еднобайтово поле, показващо типа на LLC PDU
Protocol Identification Field – петбайтово поле, съдържащо трибайтовото поле OUI и двубайтово поле Type, за идентификация на протокола от по-високо ниво, за който е предназначен кадъра
Data – съдържа данни с размер от 38 до 1492 байта
FCS – осигурява проверка на правилността на предаденият кадър и е с размер от четири байта

6.7.2. MAC подслой

Този подслой контролира достъпа до комуникационната среда по която се извършва предаването на физическите сигнали, осъществява адресацията, формира кадри със съответните полета. При предаване MAC-подслоя получава от LLC подслоя съответен LLC – блок данни, който включва в полето за данни на съответния кадър, добавя към него адресна информация, след което кодира тези полета с шумоустойчив код и записва полученото контролно значение в контролното поле. Формирането на кадъра завършва с поставянето на полетата за начало и край и предаване на кадъра към физическия слой, който го доставя във вид на неструктуриран поток от битове до възела-получател. Неговият MAC-подслой, осигурява разпознаване на MAC-адреса, открива грешките в кадъра, възникнали при предаването му, отделя данните от кадъра и ги предава към горния LLC слой.

6.8.Стандарт IEEE 802.3 (Ethernet)

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=18**

Този стандарт описва LAN с логическа топология тип “шина”, докато физическата варира от тип „линейна шина“, в по-старата версия, до тип „звезда“ в новите версии. Този тип мрежа е разработена през 60-те години на миналия век и е най-популярната LAN архитектура в момента. Скоростта на предаване за първоначалния стандарт (Ethernet) е 10 Mbps, а в последните версии на стандарта - 100 Mbps (Fast Ethernet), 1 Gbps (Gigabit Ethernet), 10 Gbps (10 Gigabit Ethernet), 40 Gbps (40 Gigabit Ethernet), 100 Gbps (100 Gigabit Ethernet).

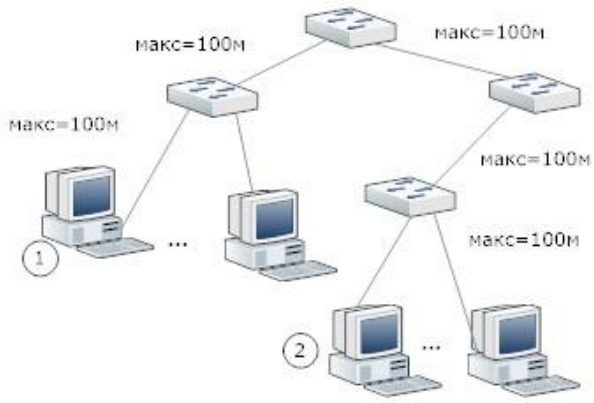
При изписването на стандарта се следва последователността:

- скорост на предаване на сигнала изразена в Mbps.
- метод на предаване, имащ две значения: ***Base*** – за директно цифрово предаване; ***Broad*** – за модулирано аналогово предаване;
- дължина на сегмента в стотици метри. Ако L заема стойности {T, TX, T4} се използва кабел с усукани двойки проводници, а при стойности {F, FX} се използва влакнесто-оптичен кабел.

Примери за означения и характеристики на различните спецификации на стандарта са:

- Ethernet

- 10BROAD36 (IEEE 802.3b) – използва CATV (Cable Television) технологията с физическа топология линейна шина или дърво. Дължината на сегмента е до 3600 м. Метод на предаване – дуплекс. Използва по един кабел във всяка посока или един кабел и честотно деление. Необходимата ширина на честотната лента за всяка посока е 18 MHz (14 MHz за данни и 4 MHz за обслужване на колизии) т.е. 36 MHz за поддържане на двете трасета;
- 10BASE5 (IEEE 802.3) - използва дебел коаксиален кабел и физическа топология линейна шина. Дължината на сегмента е до 500 м. Максимален брой възли за сегмент е 100. Метод на предаване – полудуплекс;
- 10BASE2 (IEEE 802.3a) - използва тънък коаксиален кабел и физическа топология линейна шина. Дължината на сегмента е до 185 м. Максимален брой възли за сегмент е 30. Метод на предаване – полудуплекс;
- 10BASE-T (IEEE 802.3i) - използва две усукани двойки проводници Категория 3 и по-висока и физическа топология тип „звезда“. Максимална дължина на покривното разстояние е 100 м. Метод на предаване – полудуплекс и дуплекс. Валидно е правилото 5-4-3, което е ограничение за по-ранните реализации на Ethernet, използващи коаксиален кабел. Според него между два комуникационни възела не може да има повече от пет мрежови сегмента свързани с 4 повторителя и не повече от 3 запълнени сегмента в мрежата (другите два сегмента служат за удължаване на разстоянието). Всеки запълнен повторител се брои за сегмент. Максимално достижимата дължина, без използването на комутатори, по това правило е 500 метра (фигура 51), между компютър 1 и компютър 2;



фигура 51 *Правило 5-4-3 при 10BASE-T*

- 10BASE-FL (IEEE 802.3j) - използва кабел с оптично влакно и физическа топология тип „звезда“ (най-често от точка до точка). Максимална дължина на покривното разстояние е 2000 м. Метод на предаване – пълен дуплекс.

- Fast Ethernet

- 100BASE-TX (IEEE 802.3u) – използва две усукани двойки проводници и физическа топология тип „звезда“. Максимална дължина на покривното разстояние е 100 м. Метод на предаване – полудуплекс и дуплекс;
- 100BASE-T4 (IEEE 802.3u) – ранна версия на Fast Ethernet стандарта. Използва четири усукани двойки проводници от категория 3 или по-висока. Една от двойките се използва за предаване, друга двойка за приемане, а останалите две двойки се използват за предаване или приемане на данни в двете посоки. Метод на предаване – полудуплекс (три от четири двойки проводници се използват в даден момент за предаване или приемане). Този подход позволява използването на по-ниска категория кабел. Необходимостта от три кабела е наложена от използваната схема за кодиране – 8B6T. MAC подслоя конвертира всеки байт в шест троични символа;
- 100BASE-FX (IEEE 802.3u) - предлага 100 Mbit/s пренос на данни по 62.5/125 μm и 50/125 μm мултимодова оптична среда. Физическата топология е тип „звезда“. Максимална дължина на покривното разстояние за сегмент е до 412 м. Метод на предаване – полудуплекс и дуплекс.

- Gigabit Ethernet

- 1000BASE-T (IEEE 802.3ab) – - предлага 1 Gbit/s пренос на данни през четири усукани двойки проводници от категория 5 или по-висока. Максимална дължина на покривното разстояние е 100 м. Метод на предаване – полудуплекс (ако се използват само 2 двойки проводници) и дуплекс;

- 1000BASE-X (IEEE 802.3z) – предлага 1 Gbit/s пренос на данни по оптични влакна. Негови разновидности са 1000BASE-SX, 1000BASE-LX, 1000BASE-LX10, 1000BASE-BX10, 1000BASE-ZX и 1000BASE-EX.

- 10 Gigabit Ethernet

- 10GBASE-T (IEEE 802.3an) - предлага 10 Gbit/s пренос на данни през четири усукани двойки проводници от категория 6 (разстояние до 55м), 6а (разстояние до 100м) или по-висока. Метод на предаване – пълен дуплекс от точка до точка;

- 10GBASE-SR, 10GBASE-LR, 10GBASE-ER, 10GBASE-SW, 10GBASE-LW, 10GBASE-EW (IEEE 802.3ae) – предлага 10 Gbit/s пренос на данни по оптични влакна. Метод на предаване – пълен дуплекс от точка до точка;

Най-разпространените стандарти в момента са: *802.3u* – работещ при 100 Mb/s и *802.3z* и *802.3ab*– работещи при 1Gb/s.

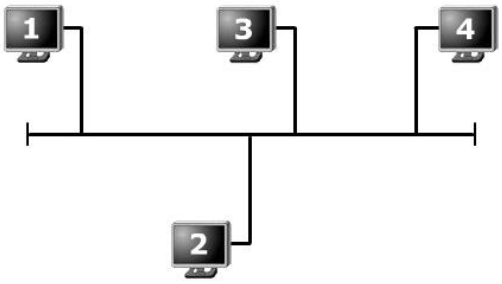
6.8.1. Методи на предаване

- Модулирано аналогово предаване (broadband)

При *модулираното предаване* се осигуряват повече от един канал за предаване по кабела. По каналите се предават аналогови сигнали от различните възли. Понеже аналоговите сигнали имат по-малко затихване и по-голяма шумоустойчивост от цифровите сигнали, покриваните разстояния са по-големи. Предаването се извършва само в едната посока на шината. Всички мрежови възли предават само към единия край на шината, на който се намира специално устройство – честотен конвертор. Той конвертира честотата f_2 на предадения сигнал, достигнал до него, в друга честотна лента f_1, в която мрежовите възли извършват приемане на сигнала. Пример може да се даде със стандарта IEEE 802.3b (10BROAD36).

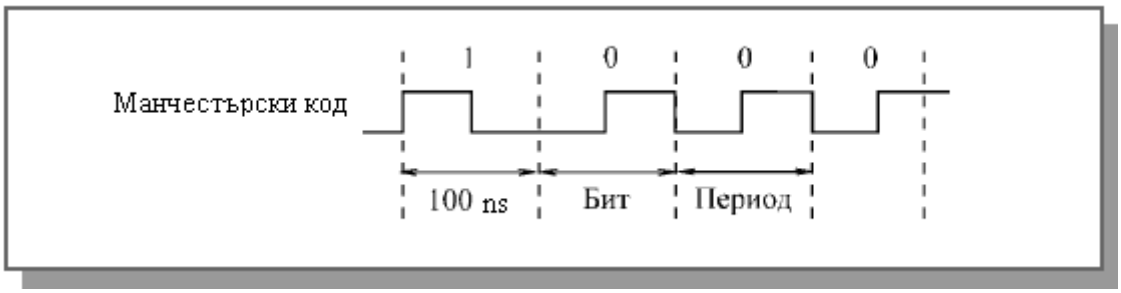
- Директно цифрово предаване (baseband)

При *директното предаване* се използва цялата честотна лента на кабела, като по този начин се осигурява само един канал за предаване по него. Например, при шинна топология (фигура 52) сигналът се разпространява едновременно в двете посоки на шината. В двата ѝ края се абсорбира от терминиращи съпротивления. Разстоянията, които се покриват са по-малки, отколкото при модулираното предаване. Сигналът е цифров, кодиран най-често с манчестърски код.

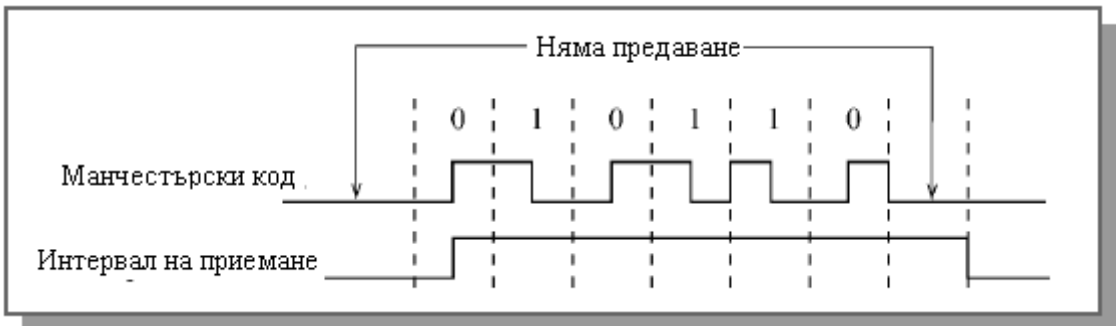


фигура 52 *Ethetnet (IEEE 802.3)*

Манчестърският код (или код Манчестър-II) е самосинхронизиращ се сигнал от две нива, което подобрява шумозащитеността и опростява приемните и предаващи възли. Периодът за предаване на един бит се разделя на две равни части. Логическа единица се кодира с високо напрежение в първия период и ниско напрежение във втория период (отрицателен преход в центъра на битовия интервал). Логическа нула се кодира с ниско напрежение в първия период и високо напрежение във втория период (положителен преход в центъра на битовия интервал). Преходът в средата на периода служи за синхронизация. Скорост на предаване 10 Mbit/s изисква лента на пропускане 10 MHz, при кодиране на 1 бит в 1 Hz (фигура 53) и период от време 100 наносекунди (10^{-9} от секундата) за честота 10 MHz и радио вълни с дължина 30 m (къси вълни).



фигура 53 *Скорост на предаване и пропускателна способност при манчестърски код*



фигура 54 *Определяне начало и край на приемане при манчестърски код*

Манчестърски код се използва както в електрически, така и във влакнесто-оптични кабели (едното ниво е отсъствие на светлина, а другото нейното наличие). Наличието на средна стойност на сигнала и отклонението от нея по време на предаването позволява откриването на колизии при Ethernet.

За Fast Ethernet този подход не може да се използва поради създаването на трудности с разпространението на високочестотни сигнали, тъй като за постигане на скорост от 100 Mbs трябва да се използва честотна лента от 200 Mz, а Категория 5e UTP позволява до 125MHz. Оптиката също се затруднява с подобна честота. За успешно предаване с висока скорост се използват други формати за кодиране. Например, стандартът 100BASE-FX използва NRZI (Non-Return-to-Zero, Invert-on-one) кодиране, а при спецификацията 100BASE-TX функционира разновидност на NRZI, наречена NRZI-3. Тези подходи дават възможност за намаляване на честотната лента и увеличаване на плътността на сигнала.

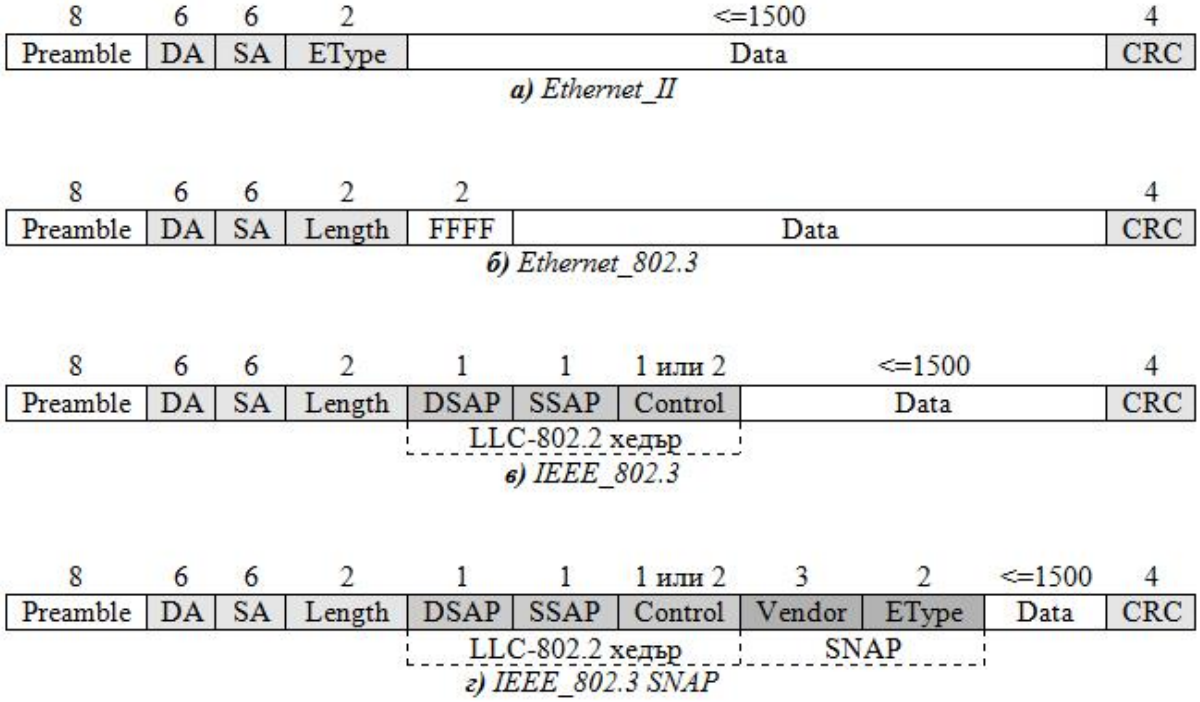
6.8.2. MAC-подслой на стандарта IEEE 802.3

За обслужване на стандартите от тип Ethernet съществуват четири различни типа кадри (проследено в исторически план):

- Ethernet_II(DIX);
- Ethernet_802.3;
- IEEE 802.3(Ethernet_802.2);
- IEEE 802.3 SNAP.

Кадрите са несъвместими, но могат да съществуват в една мрежа при условие, че се обработват от маршрутизатор. За избягване на усложнения при комуникацията е необходимо използването на еднотипни кадри в мрежата. Драйверите на мрежовите адаптери дават възможност за промяна

на подразбиращия се тип на кадъра. На следващата фигура са представени споменатите четирите типа рамки.



фигура 55 *Разновидности на Ethernet кадъра*

Предназначението на полетата е както следва:

а) Ethernet_II

Preamble – 64 бита за синхронозация (7 байта 10101010, 8-ят - 10101011)
DA – адрес на местоназначение (6 байта)
SA – адрес на източника (6 байта)
EType – 2 байта за идентификация на протокол (например 0800=IP)
Data – данни от по-горен слой (<=1500 байта)
CRC – контрол на кадъра (4 байта)

б) Ethernet_802.3

Preamble – синхронозация (8 байта)
DA – адрес на местоназначение (6 байта)
SA – адрес на източника (6 байта)
Length – 2 байта, стойност <=1500 байта
FFFF – 2 байта, нулева конторолна сума (ограничение за пренасяне само на IPX)
Data – данни от по-горен слой (<=1500 байта)
CRC – контрол на кадъра (4 байта)

в) IEEE 802.3(Ethernet_802.2)

Preamble – синхронозация (8 байта)
DA – адрес на местоназначение (6 байта)

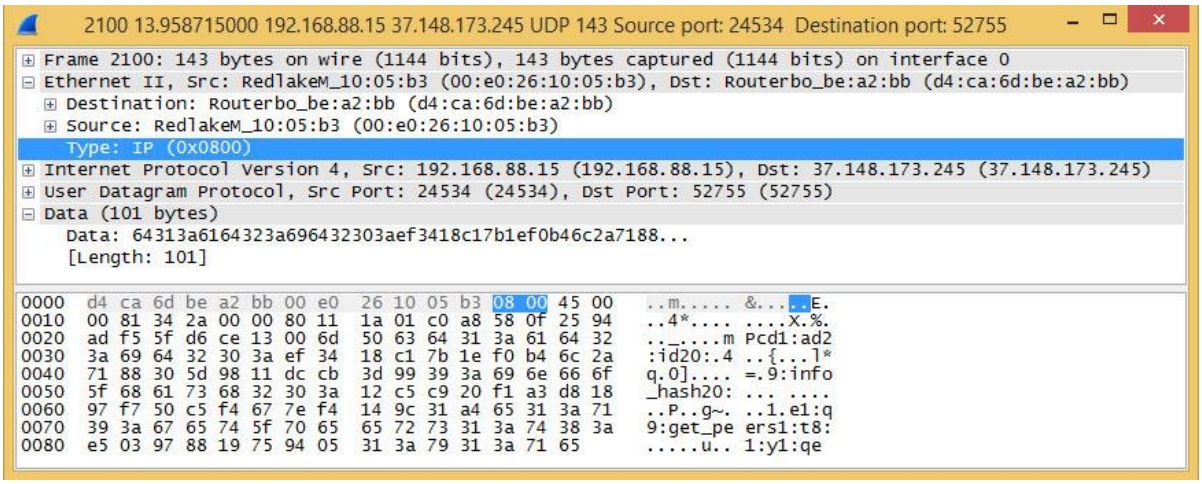
SA – адрес на източника (6 байта)
Length – 2 байта, стойност <=1500 байта
DSAP – 1 байт, точка за достъп до услугата на местоназначение
SSAP – 1 байт, точка за достъп до услугата на местоназначение
Control – LLC с установяване на конекция или без установяване на конекция (1 или 2 байта)
Data – данни и служебна информация от протоколите от по-горен слой (<=1500 байта)
CRC – контрол на кадъра (4 байта)

Полетата DSAP, SSAP и Control се групират в логически LLC-802.2 хедър.

г) IEEE 802.3 SNAP

Preamble – синхронозация (8 байта)
DA – адрес на местоназначение (6 байта)
SA – адрес на източника (6 байта)
Length – 2 байта, стойност <=1500 байта
DSAP – 1 байт, точка за достъп до услугата на местоназначение
SSAP – 1 байт, точка за достъп до услугата на местоназначение
Control – LLC с установяване на конекция или без установяване на конекция
Vendor code – 3 байта код на производителя
EType – 2 байта за идентификация на протокол
Data – данни и служебна информация от протоколите от по-горен слой (<=1500 байта)
CRC – контрол на кадъра (4 байта)

Преамбюлът, по принцип, не се приема като част от самия кадър, но предхожда всеки един такъв. В това представяне към него е добавен и разграничителят за начало на кадъра (общо 7+1=8 байта).



фигура 56 *Ethernet_II кадър*

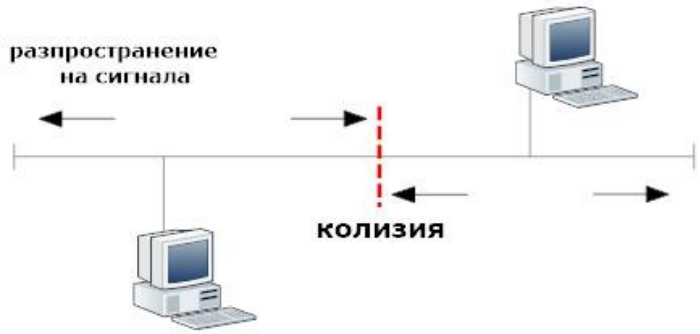
От описанието на кадрите се вижда, че Ethernet_II съдържа двубайтово поле за идентификация на протокола. Това дава възможност да се определи протоколът (например, 0x0800 за IP) от мрежово ниво (IP, ARP, RARP, IPX), чийто пакет се пренася. Пример за реален кадър от този тип може да се види на фигура 56.

Кадрите от тип Ethernet_802.3 и IEEE 802.3 не притежават такова поле, което означава, че могат да пренасят само един тип протокол (например, IP но не и IP и ARP). За TCP/IP мрежите, които използват повече от един протокол това е неприемлив вариант.

IEEE 802.3 SNAP съдържа също поле за тип, но е по-малко предпочитан защото добавя допълнителни 5 байта след LLC хедъра, за сметка на преноса на данни.

6.8.3. Множествен достъп с разпознаване на носещата и откриване на колизии (CSMA/CD)

Стандартът IEEE 802.3 използва протокол с името CSMA/CD (Carrier Sense Multiple Access With Collision Detection). Този протокол допуска, че всички възли в мрежата са равноправни, като им позволява да предават по общата комуникационна среда (шина), състезавайки се помежду си. Методът се основава на възможността всеки възел да разпознава кога шината е заета или свободна.



фигура 57 *Колизионен домейн*

След получаване на заявка за предаване от протоколите на горните слоеве, протоколът CSMA/CD формира кадър, който се предава в двете посоки по шината. В същото време друг възел може също да изпрати кадър в шината. Възниква конфликт (колизия) между двата кадъра, в следствие на което се получава деформация на сигнала. За избягването му се грижат мрежовите адаптери. При възникване на конфликт по мрежата се предава

специален заглушаващ сигнал. Всеки възел, участвал в конфликта, изисква различен интервал от време преди да изпрати отново кадъра си. След 16 неуспешни опита контролерът на мрежовата платка предава към компютъра сигнал за грешка. Общият брой на устройствата, които се съревновават за честотната лента се нарича колизионен домейн (фигура 57). Процедурата по предаване в свободна среда и тази по време на колизия може да се опише по следния начин:

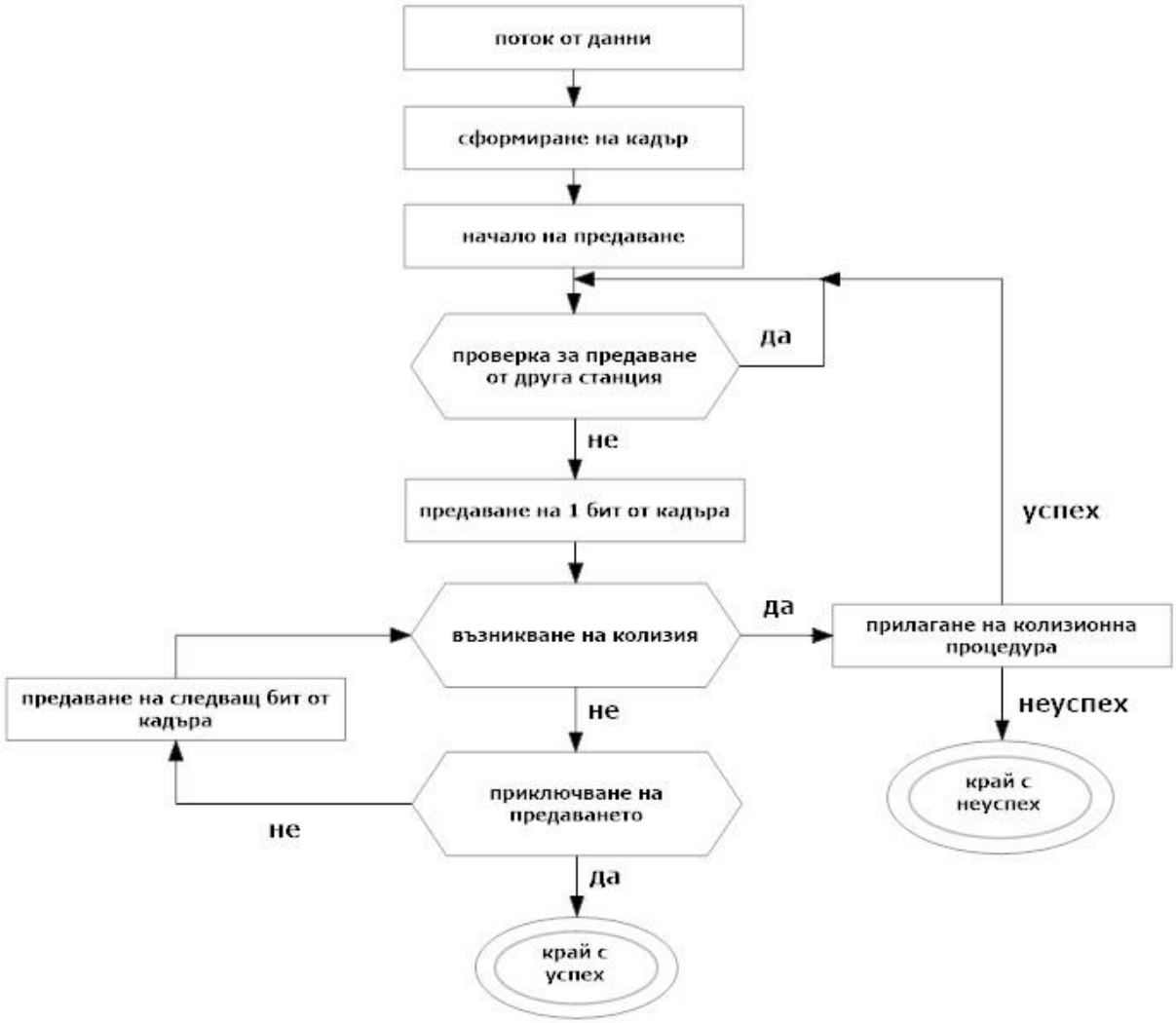
- Главна Процедура

1. Кадърът е готов за предаване;
2. Свободна ли е средата? Ако не, изчаква се до нейното освобождаване;
3. Предаване;
4. Има ли колозия? Ако да, преминаване към *колизионната процедура*;
5. Успешно предаване;

- Колизионна процедура

1. Продължаване на опита за предаването;
2. Достигнат ли е максималния брой опити? Ако да, неуспех на предаването;
3. Случайно генериран период на изчакване;
4. Преминаване към *точка 1*;

Диаграма за предаване на кадър по метода на съревнование до общата среда за комуникация е представена на фигура 58.



фигура 58 *Алгоритъм за обработка на колизии*

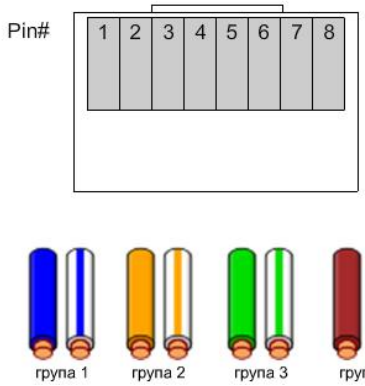
При увеличаване на възлите в мрежата конфликтите се увеличават и средната скорост намалява.

Локалните мрежи по стандарта IEEE 802.3 могат да бъдат комутирани. В този случай комуникационната среда престава да бъде обща. Използват се устройства – комутатори (устройства с високоскоростна комутационна матрица), които заместват концентраторите.

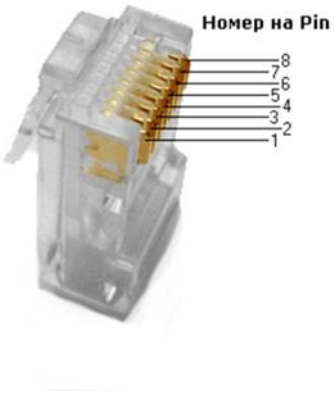
6.8.4. Използване на конектори тип RJ-45

RJ-45 (Registered Jack, фигура 59б) е физически конектор употребяван масово при съвременните локални мрежи за установяване на свързаност между устройства, на които физическите интерфейси позволяват използването на кабел с усукани двойки проводници. За целта стандартът TIA/EIA-568-B.1-2001 дефинира цветовите групи (фигура 59а), подредбата

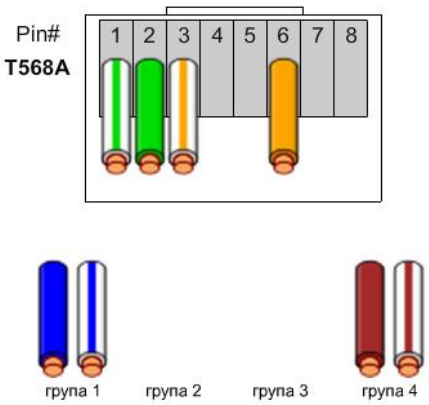
на цветовете T568A и T568B (фигура 59в,г,д,ж) и две основни схеми за реализиране на такъв тип свързаност – прав и кръстосан кабел.



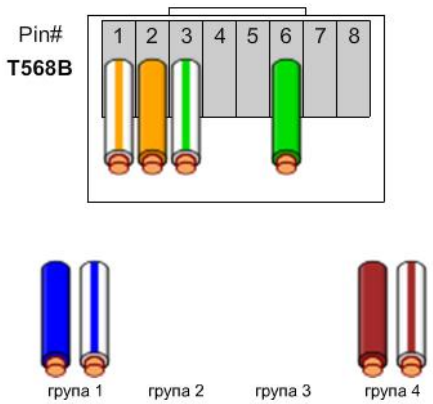
а) цветови групи



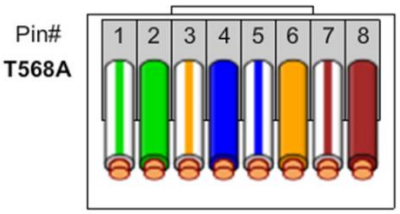
б) RJ-45 конектор



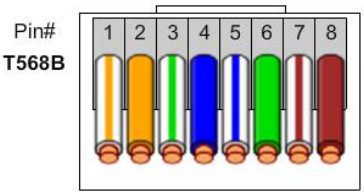
в) използване на две усукани двойки



г) използване на две усукани двойки



д) използване на четири усукани двойки



ж) използване на четири усукани двойки

фигура 59

Ако в двата края на кабела се приложи една и съща цветова схема (T568A или T568B) то получената реализация се нарича прав кабел (straight-through cable). Ако в двата края се използват различни схеми (T568A и T568B) то кабелът се нарича кръстосан (crossover cable).

Тези схеми на свързаност са продиктувани от факта, че предаващите двойки пинове на порта от едната страна трябва да бъдат свързани с приемащите такива от другата страна. Всеки порт, поддържащ този начин на свързване може да бъде означен като MDI (medium dependent interface)

или MDI-X (medium dependent interface crossover). Стандартно MDI се използва за крайните устройства (мрежови адаптери), докато MDI-X за портовете на междинните устройства (комутатори, концентратори). Правият кабел свързва MDI с MDI-X интерфейси, докато кръстосаният кабел е предназначен за еднотипни такива. Пример за използване на прав кабел може да бъде връзката между мрежов адаптер на компютър (MDI) и порт на комутатор (MDI-X). Ако поне едно от двете устройства поддържа функцията Auto-MDI/MDI-X, то може автоматично да открива необходимия вид кабелна връзка и да се преконфигурира по подходящ начин. Така се премахва необходимостта от кръстосан кабел. Съвременните междинни устройства притежават тази функция. В таблица 1 са показани MDI конфигурациите на пиновете на спецификациите 10Base-T, 100Base-TX и 1000Base-T. От нея се вижда, че 1000BASE-T използва и четирите двойки проводници за предаване в двете посоки (BiDirectional).

Pin	Сигнал	10Base-T	100Base-TX	1000Base-T
1	Предаване (+)/Двупосочен	TX+	TX+	BI_DA+
2	Предаване (-)/Двупосочен	TX-	TX-	BI_DA-
3	Приемане (+)/Двупосочен	RX+	RX+	BI_DB+
4	Липсва/Двупосочен	Липсва	Липсва	BI_DC+
5	Липсва/Двупосочен	Липсва	Липсва	BI_DC-
6	Приемане (-)/Двупосочен	RX-	RX-	BI_DB-
7	Липсва/Двупосочен	Липсва	Липсва	BI_DD+
8	Липсва/Двупосочен	Липсва	Липсва	BI_DD-

таблица 1 *MDI конфигурациите на пиновете*

6.9.Стандарт IEEE 802.4 (Token Bus)

Стандартът IEEE 802.4 е пример на архитектура с различни логическа и физическа топология. Логическата топология е кръг, а физическата е тип “шина” (фигура 60). Стандартът е създаден за посрещане на нуждите в областта на автоматизацията на производството, тъй като Ethernet и Token Ring не задоволявали изискванията на производствения процес.

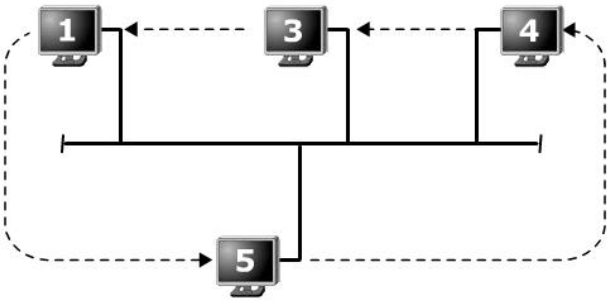
Всяка станция има определено място в общата подредба, където се спазва логическа последователност спрямо номера, който притежава. Тя знае адреса на своя ляв и десен съсед и играе активна роля в процеса на поддържане на мрежовата организация. Предаването е съобразено с наложения времеви лимит от мрежата. Кабелната среда е широколентов

коаксиален кабел със съпротивление 75 Ω. Използваните режими на предаване са немодулирано аналогово предаване (carrierband) и модулирано аналогово предаване. При немодулираното аналогово предаване се използва един канал със скорост 5 Mbps. Дължината на сегмента може да достигне до 700 м с максимален брой възли – 32. При модулираното аналогово предаване се реализират няколко 10 Mbps канала за връзка.

6.9.1. MAC-подслой на IEEE 802.4

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=19**

Действащият протокол за стандарта е с наименованието Token Bus. При него за достъп до комуникационната линия се използва управляващ маркер (token). Това е специален кадър, разрешаващ достъпа до шината, който се предава от станция към станция. Единствено тази, в която е маркерът, има право да предава, като по този начин се премахва възможността за конфликт. Всеки възел знае адреса на съседа си отляво и отдясно. При инициализиране на мрежата право да предава получава станцията с най-голям адрес. Тя изпраща своя кадър, след което предава маркера към възела със следващия по-малък адрес. На практика маркерът достига до всички възли по шината, но само станцията с адрес указан в маркера, го прехваща.



фигура 60 *Стандарт IEEE 802.4*

Всеки възел владее маркера само за определен период от време, през който изпраща кадрите си. Ако някоя станция няма данни за предаването тя препредава маркера на следващата, определена от общата подредба. В този стандарт са дефинирани четири класа на приоритетност – 0, 2, 4 и 6, където 0 е най-ниския, а 6 е най-високия приоритет. Всеки възел поддържа четири опашки за кадрите с различен приоритет. Новосформираният кадър получава стойност за приоритет и се подрежда в съответната опашка. Когато станцията получи маркера, кадрите от различните опашки се

предават в строго определен ред. За всяка опашка съществува времеви интервал, през който да предава. При липса на кадри маркерът се предава на следващата опашка. При достигане на ниво 0 и липса на кадри, маркерът се прехвърля към следващата станция в логическия кръг. Процесът се контролира от таймери за всяко приоритетно ниво. Приоритетната схема гарантира за ниво 6 част от честотната лента., което позволява нейното използване за контролиране на системата в реално време. Например, ако имаме 10 Mbps свързаност, 10 станции и конфигурация, дефиниращa 1/3 от честотната лента за трафика от ниво 6 то за всяка станция има гарантирани 0,33 Mbps скорост.

Форматът на кадъра е представен на фигура 61.

фигура 61 *Кадър на протокола Token Bus*

Preamble – синхронозация (1 байт)
Start – 1 байт, начало на кадър
Control – 1 байт, определя типа на кадъра като даннов или контролен. При даннов кадър полето включва приоритетното ниво и индикатор за коректно или некоректно получаване от страна на станцията-получател. При контролен кадър полето определя типа на кадъра.
DA – хардуерен адрес на станцията-получател (2 или 6 байта)
SA – хардуерен адрес на източника (2 или 6 байта)
Data – данни (<=8182 байта при 2 байта за адрес, <=8174 байта при 6 байта за адрес)
CRC – контрол на кадъра (4 байта)
End – 1 байт, край на кадър

Типовете контролни кадри са показани в таблица 2.

Контролно поле (Control)	Наименование	Значение
00000000	Claim_token	Заявка за маркер при инициализация на кръга
00000001	Solicit_successor_1	Подкана за станция да се включи към виртуалния кръг
00000010	Solicit_successor_2	Подкана за станция да се включи към виртуалния кръг
00000011	Who_follows	Възстановяване от загубен маркер
00000100	Resolve_contention	Предава се при опит на повече от една станция да се включат към кръга
00001000	Token	Пускане на маркер
00001100	Set_successor	Установяване на нов наследник

таблица 2 *Типове контролни кадри*

Периодично активната станция предава Solicit_successor кадър за подкана за включване на нова станция към кръга. Кадърът съдържа адреса на изпращача и на неговия наследник. Само станциите с адрес между посочените могат да се включат към кръга. Ако липсват кандидати мрежата възстановява нормалната си работа. При наличието на един кандидат, при неговото включване, той автоматично става наследник на активната станция в логическия кръг. При опит за включване на повече кандидати се получава сблъсък, които се решава чрез стартиране на арбитражен процес (чрез броадкаст Resolve_contention) от страна на станцията-притежател на маркера, който определя по случаен начин изчакващи периоди за включване (всеки интерфейс поддържа такава възможност).

При натоварен трафик (следи се чрез таймер във всяка станция за пристигане на маркера) не се отправят покани за включване.

При напускане на станция тя предава на съседа си с по-малък номер, че наследникът му вече е съседът ѝ с по-голям номер чрез Set_successor.

Инициализацията на кръга е специален случай за добавяне на нова станция. При включване на първата станция към кръга, тя регистрира, че липсва трафик и изпраща Claim_token кадър. При липса на отговор инициализира кръг с един участник и периодично отправя покани за нови станции.

Възможно е възникването на проблеми, свързани с маркера и виртуалния кръг. За избягването им след предаване на маркера станцията следи дали наследника предава. Ако липсва активност тя генерира втори маркер. При нов неуспех станцията изпраща Who_follows кадър като посочва своя наследник. Когато наследникът на пропадналата станция получи този кадър той отговаря със Set_successor като задава себе си за нов наследник. Пропадналата станция се премахва от кръга. Ако две последователни станции отпаднат от кръга Who_follows не връща отговор. В тази ситуация станцията изпраща Solicit_successor_2 кадър за да провери дали има активни станции.

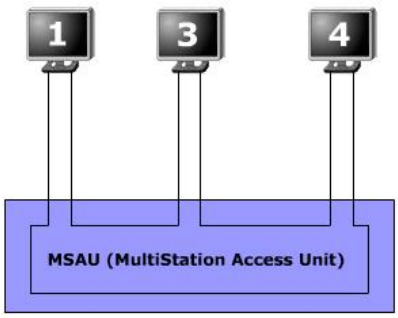
6.10. Стандарт IEEE 802.5

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=20**

Изграден е на базата на стандарта Token Ring, създаден от фирмата IBM през 1970 година. Подобно на IEEE 802.4 развитието му е прекратено от IEEE организацията. Token-Ring използва метод за достъп до средата, наречен token-passing (предаване на маркер), където в даден момент в мрежата може да съществува само един маркер. Кадрите се предават в една посока без риск от колизии, понеже само притежателят на маркера може да предава данни. Маркерът представлява 3-байтов кадър циркулиращ по кръга. Логическа топология е тип “кръг”. Физическата топология при по-старите реализации е кръг, а новите се базират на MSAU концентратори т.е. звезда () . Сигналът обхожда последователно в кръг всички нейни възли. Разстоянията, които се покриват са по-големи от тези в стандартите IEEE 802.3 и 802.4, тъй като всеки възел, през който преминава сигнала действа като усилвател за него. Могат да се използват и трите типа кабели.

6.10.1. Характеристики

- Всяка станция се свързва към централен концентратор (MSAU). Целта на MSAU е да запази функционалността на кръга чрез електрическо игнориране на неработещи устройства (когато станция е изключена или блокирала).
- Всеки мрежови адаптер работи като напълно функционален повторител (еднопосочен) - регенерира сигнала и извършва побитово повторение.
- Работи на скорост от 16 Mbps или 4 Mbps, но не и на двете едновременно (зависи от конфигурацията/типа на мрежовия адаптер). Ако съществуват разногласия относно скоростта, то кръга работи на 4 Mbps.



фигура 62 *Стандарт Token Ring, използващ централен концентратор (MSAU)*

6.10.2. MAC-подслой на стандарта IEEE 802.5

Token-passing мрежите предвижват малък кадър, наречен маркер, по кръга. Притежанието на маркера дава право за предаване на данни. Ако станцията получила маркера няма какво да предава маркерът преминава към следващата станция в потока, като всяка от тях може да задържи маркера за определен период от време (Token Holding Timer механизъм).

Станция получила маркера и имаща данни за трансфер, го задържа и променя един бит в него, като го превръща в начало на кадър с данни, добавя данните, които трябва да се предаде и изпраща кадъра към следващата станция по кръга. Докато информационният кадър се предвижва по кръга няма маркер в мрежата (освен ако не се използва ранно освобождаване на маркера), което означава, че останалите станции имащи данни за предаване изчакват до освобождаване на маркера. Информационният кадър обикаля по кръга докато достигне станцията-местоназначение, която го копира за обработка и го маркира като прочетен. Кадърът продължава да се движи по кръга докато достигне до станцията-изпращач, където се проверява дали е прочетен, след което го премахва и освобождава нов маркер.

Ако се поддържа ранно освобождаване на маркера, след предаването на информационния кадър станцията-изпращач освобождава нов маркер, който следва кадъра с информация. Въпреки това, в даден момент има само един маркер по кръга.

За разлика от CSMA/CD мрежите (като Ethernet) token-passing мрежите се определят като deterministic, което на практика означава, че може да се изчисли максималният период от време, който ще мине преди всяка станция да може да предава. Поради тази причина Token-Ring мрежите са идеални за приложения, при които закъснението трябва да е предвидимо и надеждността на мрежовите операции е от първостепенно значение.

В Token-Ring мрежите се използва приоритетна система, позволяваща на определени станции (избрани от администратора) да използват по-често мрежата. Всяко устройство има стойност на приоритет, като колкото по-голяма е тази стойност, толкова по-често въпросният хост може да използва мрежата. Token-Ring кадрите имат две полета контролиращи приоритета, това са priority и reservation полетата.

Само станции с приоритет по-голям или равен на приоритета съдържащ се в маркера могат да го задържат и да предават данни. След като маркерът е задържан и променен в информационен кадър, само станция с приоритет по-висок от този на изпращащата, може да си резервира маркера за следващото му преминаване по кръга. Станцията, която увеличава стойността на приоритет на маркера е задължена да върне тази стойност в първоначалното и положение, след като приключи с предаването на данните.

Мрежите изградени на база на технологията Token-Ring използват няколко механизма за засичане и справяне с грешките възникващи по време на работа. Една от станциите в мрежата се определя за active monitor. Тази станция (произволно избрана) действа като централен източник на синхронизираща информация за всички останали в кръга и изпълнява различни функции за управление. Например, една от тези функции е отстраняването на кадри, обикалящи продължително време по кръга. Това се получава, когато станцията източник на данните отпадне от мрежата по различни причини (изключване, блокиране и др.), преди да отстрани информационния кадър, който е излъчила. При тази ситуация кадърът би продължил да се движи по кръга вечно, като ”заключи” мрежата за използване от другите станции. Станцията определена за active monitor има за задача да засича и премахва такива кадри, както и да освобождава нов маркер.

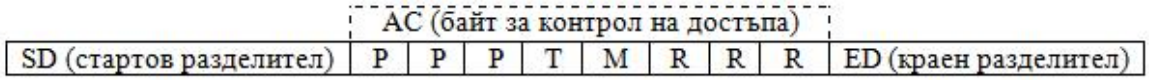
Token-Ring използва алгоритъм наречен beaconing за намиране и отстраняване на грешки. Когато станция засече сериозен проблем в мрежата (например прекъснат кабел), тя изпраща специален кадър наречен beacon frame, имащ за цел да определи областта на ”пропадане” на мрежата. Тази област обхваща всичко между станцията сигнализираща за пропадането и най-близкият и активен съсед нагоре в потока. Алгоритъмът стартира процес на самоконфигурация (autoconfiguration), при който крайните устройства от отпадналата област изпълняват автоматично диагностични операции и се опитват да преконфигурират мрежата с цел заобикаляне на отпадналата част. MSAU могат да постигнат това чрез електрическо преконфигуриране (игнориране на отпадналите станции).

Token-Ring поддържа следните типове кадри – маркер (Token frame), даннов кадър (Data frame), LLC кадър (LLC data frame), MAC кадър (MAC management frame) и прекъсващ кадър (Abort frame).

Маркерът е с дължина 3 байта и се състои от стартов разделител, байт за контрол на достъпа и краен разделител (фигура 63).

Станциите идентифицират сигнала като маркер, като проверяват състоянието на маркерния бит в полето за контрол на достъпа. Ако битът е 0 тогава това е кадър с маркер, ако е 1 - това е кадър с данни/команди.

Полета на маркера са представени на фигура 63.

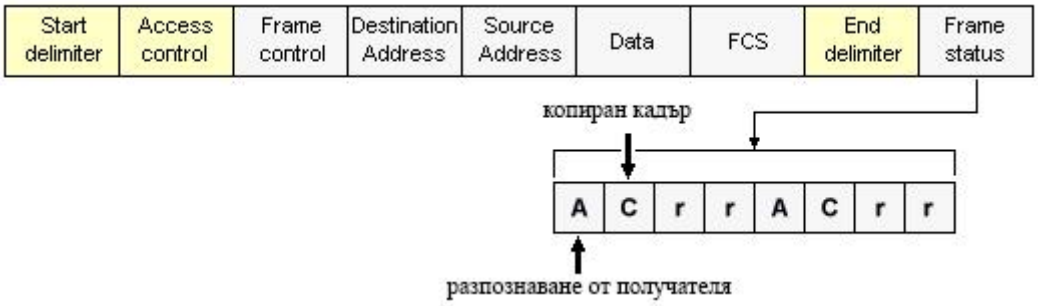


фигура 63 *Маркер на IEEE 802.5*

SD (стартов разделител) - предупреждава станцията за пристигане на кадър (маркер или даннов/команден)
AC (байт за контрол на достъпа) - съдържа поле за приоритет (3-те бита с най-голяма тежест), както и поле за резервация (3-те бита с най-малка тежест). Между тях се намират token бита (използван за разграничаване на маркера от кадър с данни/команда) и monitor бита (използван от станцията active monitor, за да определи дали кадърът обикаля безкрайно по кръга).
ED (краен разделител) - поредица от битове в края на маркера или даннов/команден кадър, указващ края на кадъра. Съдържа също и битове използвани за маркиране на ”развалени” кадри и определящи дали това е последен кадър от логическа последователност.

Данновият кадър е с различна дължина, в зависимост от големината на информационното поле, която се определя от скоростта на предаване на сигнала. Целта на този тип кадри е да пренасят информация, предназначена за протоколи от по-горен слой.

Полета на даннов/команден кадър са показани на фигура 64.



фигура 64 *Даннов/контролен кадър*

SD (стартов разделител) - предупреждава станцията за пристигане на кадър (маркер или даннов/команден). Включва сигнали, който разграничават байта от останалата част от кадъра.
AC (байт за контрол на достъпа) - съдържа поле за приоритет (3-те бита с най-голяма тежест), както и поле за резервация (3-те бита с най-малка тежест). Между тях се намират token бита (използван за разграничаване на маркера от кадър с данни/команда) и monitor бита (използван от станцията active monitor, за да определи дали кадърът обикаля безкрайно по кръга).
FC (байт за контрол на кадъра) - показва дали кадърът съдържа данни или контролна информация. В контролните кадри този байт определя типа на контролната информация.
DA и SA (адрес на местоназначение и адрес на източник) - две 6 байтови адресни полета, определящи източника и крайния получател на кадъра.
DATA (Данни) - дължината на полето зависи пряко от времето, за което станцията може да задържи маркера. Съдържа данни предназначени за протоколи от по-горни слоеве.
FCS (проверка на последователността на кадрите) - съдържа стойност попълнена от станцията-изпращач (след изпълняване на алгоритъма Frame-Check Sequence) и зависеща от съдържанието на кадъра. Станцията-получател изпълнява същия алгоритъм и сравнява стойностите, за да определи дали кадърът е пострадал по време на пътуването му.
ED (краен разделител) - поредица от битове в края на маркера или даннов/команден кадър, указващ края на кадъра. Съдържа също и битове използвани за маркиране на ”развалени” кадри и определящи дали това е последен кадър от логическа последователност.
FS (статус на кадъра) - еднобайтово поле в края на кадъра, съдържащо индикатор за разпознаване на адрес (показващ дали местоназначението е ”познало”, че кадърът е за него) и индикатор за копиране на кадъра (показващ дали местоназначението е копирало кадъра). Полетата са дублирани.

Останалите кадри са командни и съдържат управляваща информация. Всяка работна станция поддържа четири процеса за управление на мрежата – Active Monitor (AM), Ring Error Monitor (REM), Configuration Report Server (CRS) и Ring Parameter Server (RPS). Прекъсващата рамка се състои от разграничител за начало и край (общо 2 байта) и се използва за прекъсване на предаването.

6.11. Стандарт FDDI (Fiber Distributed Data Interface)

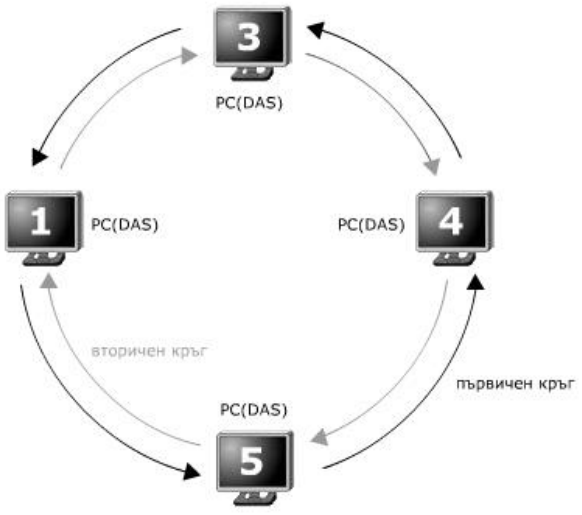
Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=21**

FDDI (Fibre Distributed Data Interface) е технология разработена от ANSI в средата на 80-те години [1][2]. Скоростта на предаване е 100 Mbps. Кабелът е оптичен. Протоколът е базиран на протокола Token Ring. Физическата топология е тип „двоен кръг“. FDDI може да се използва като опорна мрежа за свързване на няколко LAN помежду им, базирани на

Ethernet и Token Ring. Макар, че стандартът ANSI X3T9.5 определя скорост за трансфер от 100 Mbps, съществуват и спецификации надграждащи типичния FDDI и позволяващи далеч по-голяма скорост (1Gbps, 10Gbps).Появата на Fast Ethernet и Gigabit Ethernet, сложността на управление на станцията при FDDI и високата му цена са причини FDDI да не се утвърди на пазара за LAN.

6.11.1. Физически слой на FDDI

Двойният кръг на стандарта условно се разделя на два кръга обозначени като първичен (primary) и вторичен (secondary), с противоположно движещ се трафик. При нормално функциониране на мрежата се използва първичният кръг, а вторият е неизползваем (в състояние idle). Целта на втория кръг е да осигури надеждност по трансфера на данните при отказ на първичния кръг. Междинните устройства определят в коя част от мрежата е загубена връзката и автоматично свързват двата кръга. Процесът се нарича wrapping. Посоката на предаване е различна за двата кръга.Стандартното окабеляване е оптика. На по-късен етап е добавен и кабел от типа усукани двойки Категория 5, който се означава като CDDI (Copper Distributed Data Interface).



фигура 65 *Fibre Distributed Data Interface*

Покривното разстояние е различно за различните преносни среди. При представяне на оптиката е необходимо да се отбележат и двата режима на предаване: многоточков и едноточков, тъй като оказват влияние на разстоянието. Многоточковият режим дефинира максимално разстояние

между станциите до 2 километра, докато едноточковият режим до 60 километра.

Много важен момент за общата дължина на кабела е преминаването от двоен в единичен кръг. Например, при многоточков режим общата дължина на кабела не трябва да надвишава 200 километра, което означава, че за правилното функциониране на кръга, дори при повреда, е необходимо тази дължина да е наполовина (100 км).

Необходимо е да се дефинира и понятието обща дължина на кабела, като сбор на дължините на всички кабели от основния кръг и свързващите звена на станциите.

Ограничението, което се налага от стандарта е до 1000 физически връзки и е съобразено с общото закъснение на мрежата. Като се вземе под внимание факта, че всяко устройство при двойно свързване се отчита като две такива следва, че максималния брой устройства трябва да бъде 500.

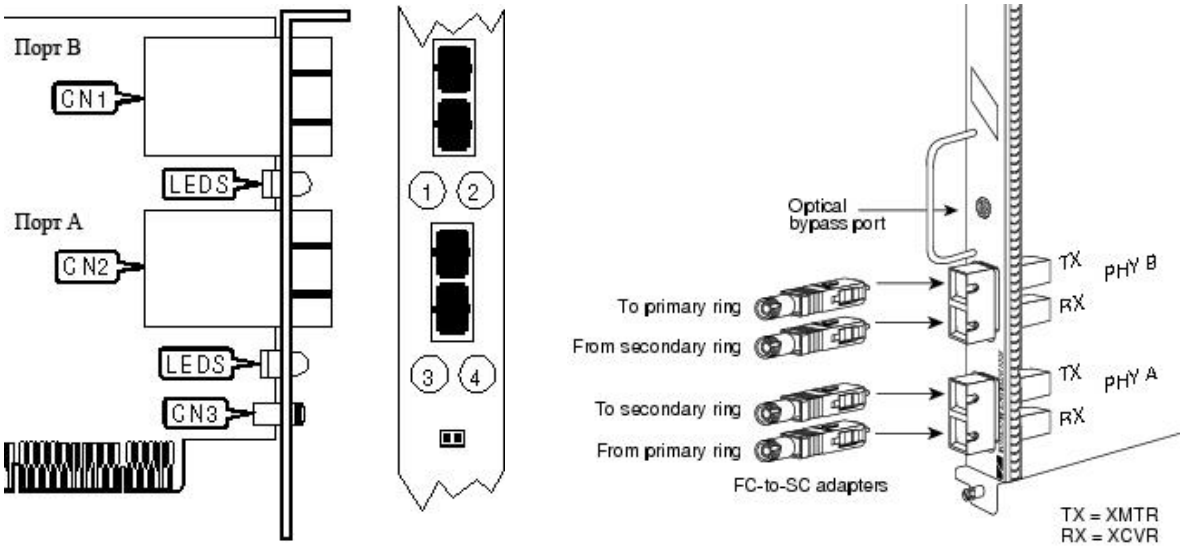
FDDI дефинира четири различни типа портове:

- Порт A – за него първичният кръг е входящ, а вторичният изходящ;
- Порт B – за него първичният кръг е изходящ, а вторичният входящ;
- Порт M – главен концентриращ порт;
- Порт S – Подчинен порт за единично свързани устройства.

На базата на тези портове могат да се реализират два начина на свързване на устройствата към мрежата – двустранно и едностранно. Спрямо начина на свързване устройствата са:

- възли, свързани към двата кръга (DAS – dual-attachment station);
- двоен концентратор (DAC – dual-attachment concentrator);
- възли, свързани само с първичния кръг (SAS – single-attachment station);
- единичен концентратор (SAC – single-attachment concentrator).

Двойните възли разполагат с по един интерфейс за свързване към всеки от кръга. Портовете към тези интерфейси се означават с A и B. Всеки от тях има по два физически конектора, което означава, че общият им брой е четири. На фигура 66 са показани два изгледа на интерфейсни платки от споменатия вид.



фигура 66 *DAS интерфейсни платки*

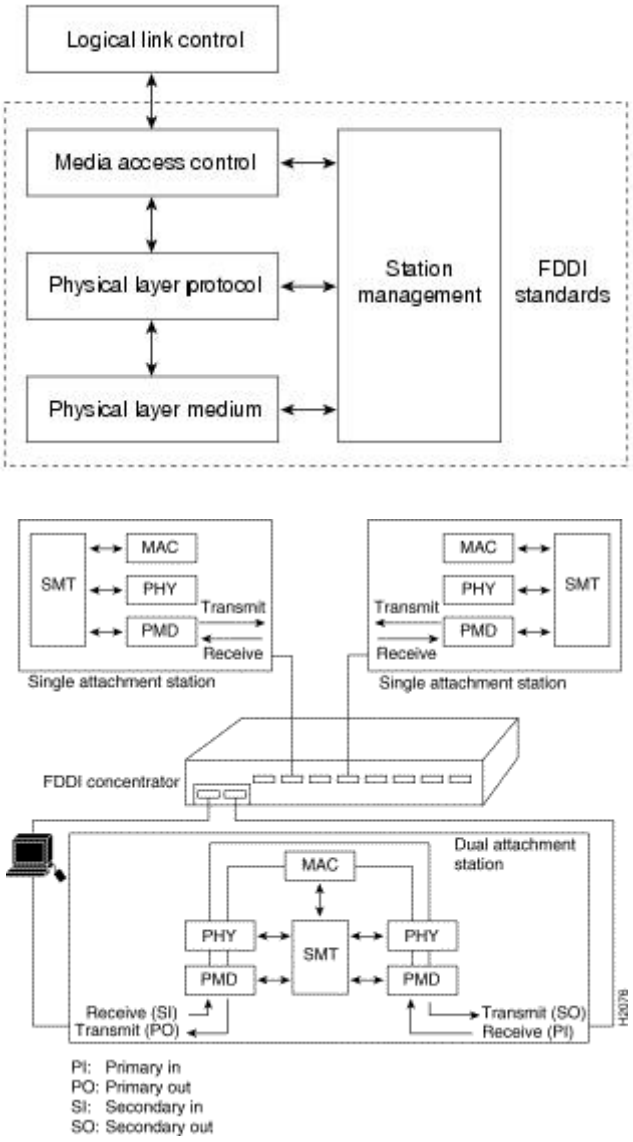
Съществуват валидни схеми за свързване на посочените портове:

- (A, B), (B, A) – за свързване на DAS устройство към двоен кръг;
- (A, M), (M,A) - за свързване на DAS устройство към концентратор при двойно свързване;
- (B, M) , (M,B) - за свързване на DAS устройство към концентратор при двойно свързване;
- (M, S) , (S, M) - за свързване на SAS устройство към концентратор;
- (S, S) – за осъществяване на връзка от точка до точка между SAS устройства.

FDDI спецификациите (фигура 67) Media Access Control (MAC), Physical Layer Protocol (PHY), Physical-Medium Dependent (PMD) и Station Management (SMT) обхващат физическия слой и MAC подслоя на Data link нивото от OSI модела.

- MAC - определя метода за достъп до медията, формат на кадъра, максималния период от време, за който дадена станция може да задържи маркера, адресиране на кадрите, алгоритми за изчисляване на CRC стойността (циклична проверка с остатък), механизми за откриване и коригиране на грешки възникващи по време на работа.
- PHY - определя процедурите по кодиране/декодиране на данните, синхронизацията на станциите и други допълнителни функции.
- PMD - определя характеристиките на използваната медия, включително вида на оптичните влакна, енергийни нива, оптични компоненти, конектори.

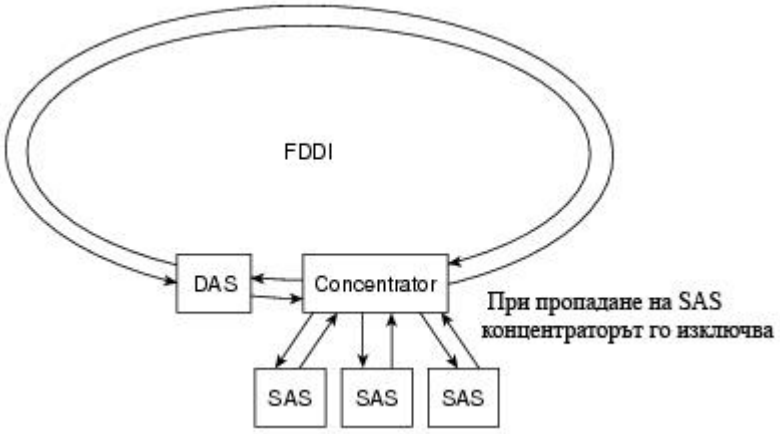
- SMT - определя конфигурацията на станциите, конфигурацията на кръга, контролни елементи на кръга, добавяне или изваждане на станции от мрежата, инициализация на устройствата, откриване и поправяне на грешки, водене на статистика.



фигура 67 *FDDI спецификации*

При прекъсване на кабелите в дадена отсечка, в зависимост от това къде се намира повредата, може да се очаква DAS устройството да затвори

кръга или концентраторът да изолира портовете на съответното SAS устройство.

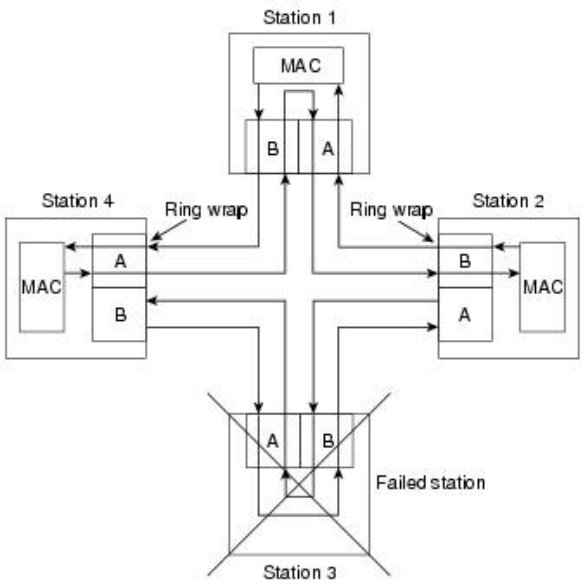


фигура 68 *Обща схема на свързване на DAS и SAS станции*

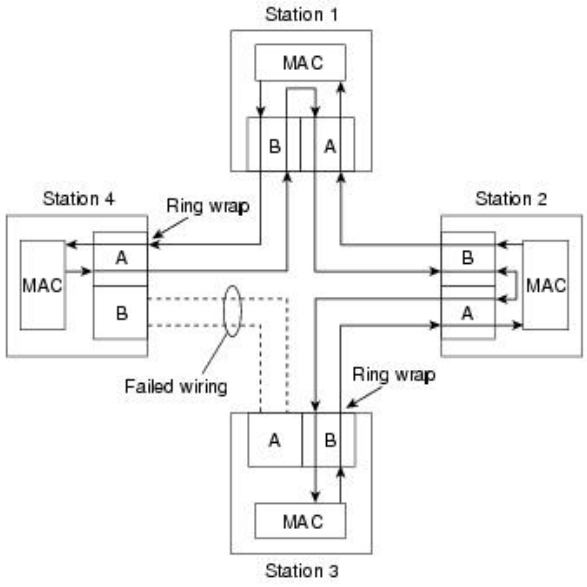
Главната характеристика на FDDI, която е пряко свързана с надеждната работа на мрежата е физическата топология двоен кръг. Когато станция от мрежата отпадне (фигура 69а) или кабелът е прекъснат (фигура 69б), кръгът се обвива (данните се маршрутизират по вторият кръг, по точно по част от него, като топологията се променя в единичен кръг). Функционалността на мрежата се запазва, но само при положение, че имаме една отпаднала DAS, ако са две или повече се образуват два (или няколко) кръга, които продължават да работят, но нямат връзка по между си.

С цел продължаване на работата на мрежата при отпадане на станция, която е част и от двата кръга се използват оптични суичове (нямащи нищо общо с устройствата използвани за комутация на пакети, по скоро са оптични превключватели) (фигура 69в). При нормална операция на мрежата този превключвател предава светлинните импулси към DAS. Ако въпросната станция не функционира, превключвателя пропуска светлината през себе си (чрез система от вътрешни огледала) осигурявайки продължаване на работа на мрежата. Предимството на използването на тези превключватели е, че кръга не изпада в състояние на обвиване при отпадането на станцията, пред която е монтиран.

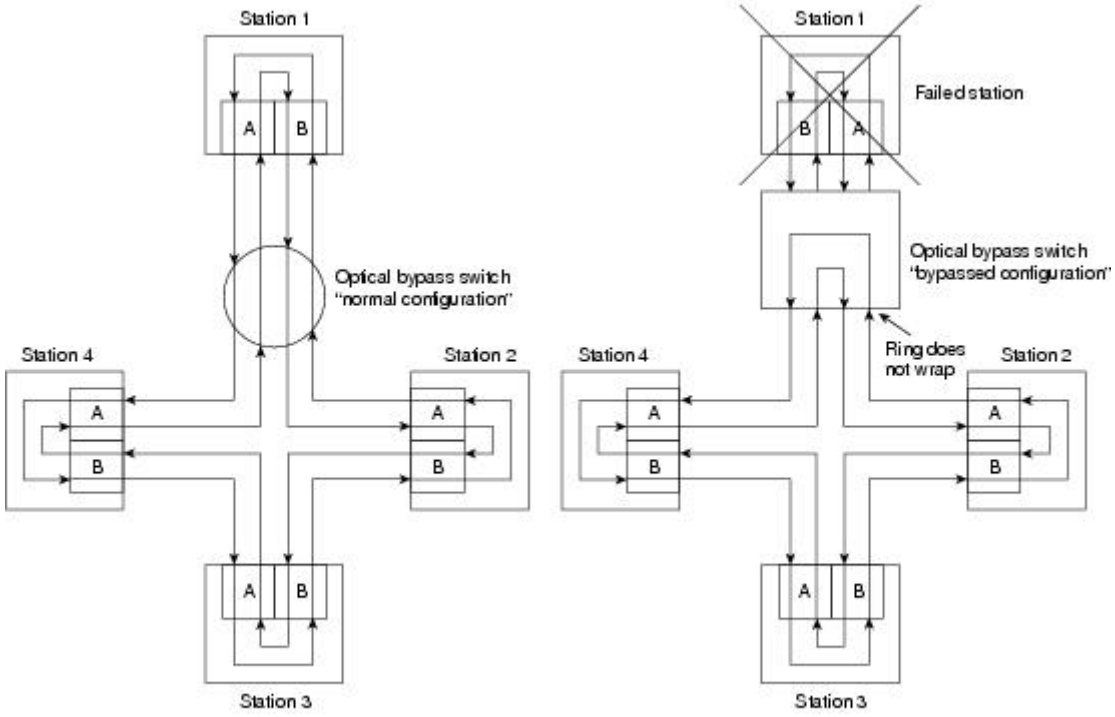
На фигура 69 са илюстрирани (от CISCO) описаните по-горе ситуации.



а) повреда на станция



б) повреда на връзката



в) използване на оптични суичове

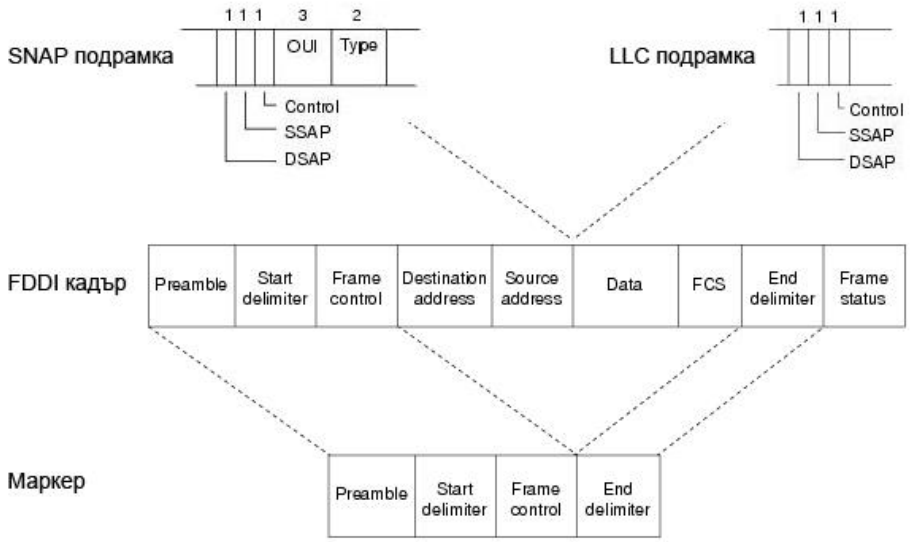
фигура 69 *Повреди при FDDI*

6.11.2. MAC подслой

Логическата топология е кръг. Протоколът е базиран на протокола Token Ring. Форматът на кадрите в FDDI е доста близък до този на Token Ring мрежите, но естествено има и съществени различия. Основното предимство на кадрите е тяхната големина - мрежата работи нормално и при кадри дълги 4500 байта. Видовете кадри са:

- FDDI кадър;
- LLC кадър – поддържа спецификацията IEEE 802.2;
- LLC SNAP кадър – поддържа LLC SNAP кадри;
- Маркер;
- SMT кадри – управляват функционирането на станциите.

На фигура 70 са показани форматите на FDDI кадър, маркер и позициите на допълнителните подрамки. Пунктираните линии обозначават общите полета, което дава добра визуална представа за позицията на вмъкване на допълнителните полета. SNAP и LLC подрамката се вмъкват между Source address и Data полетата на FDDI кадъра, като добавят указаната информация.

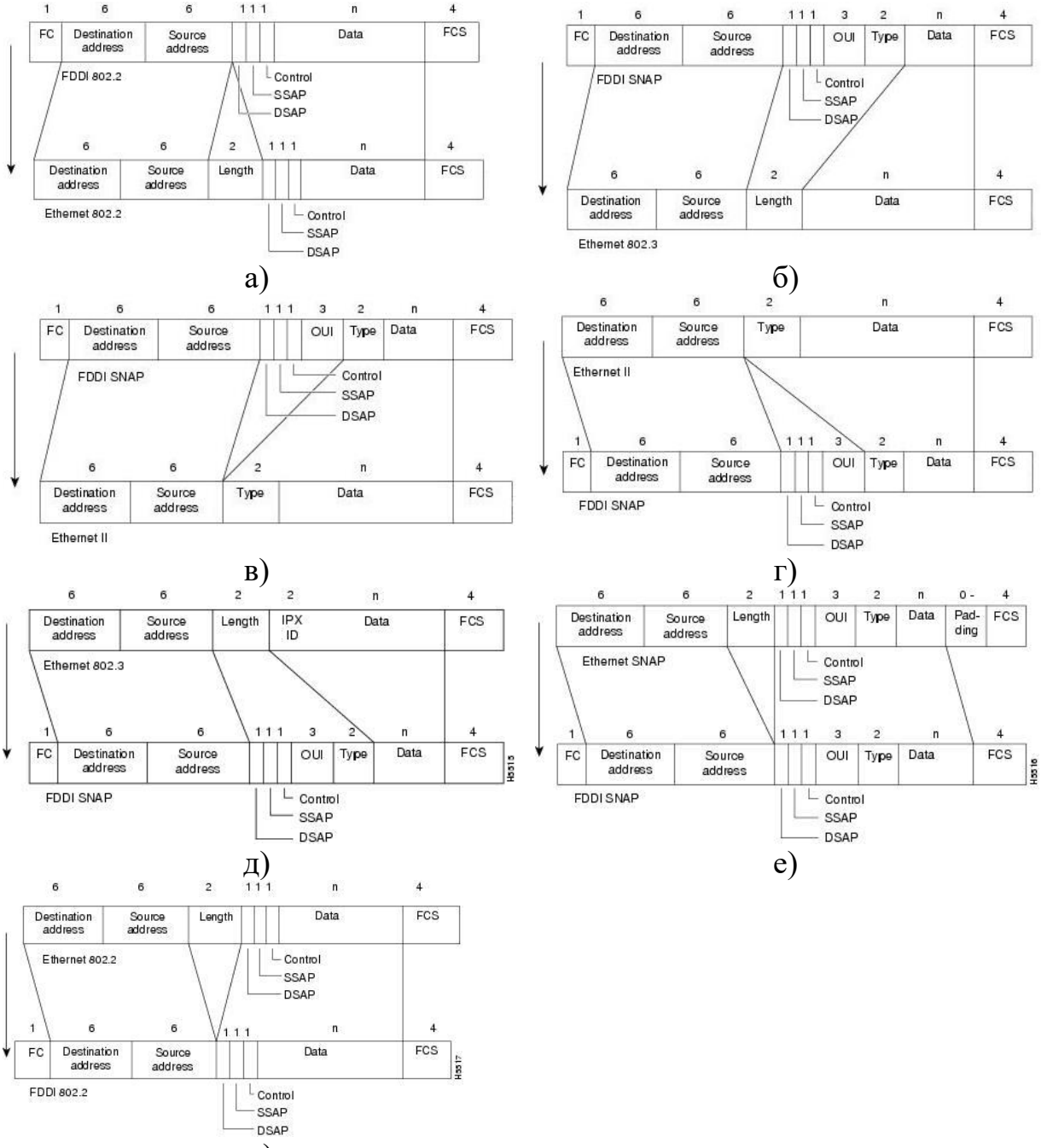


фигура 70. *Видове кадри за данни при FDDI*

PR (Preamble) - уникална последователност от битове подготвяща станцията за приемане на кадъра.
SD (Start Delimiter) - обозначава началото на кадъра, съдържа сигнални елементи различаващи полето от останалата част от кадъра.
FC (Frame Control) - показва дължината на адресните полета и дали кадъра съдържа синхронни или асинхронни данни, пренася и контролна информация.
DA (Destination Addres) - може да съдържа unicast, multicast или broadcast адрес.
SA (Source Address) - съдържа адреса на станцията изпращаща кадъра.
DATA - съдържа или контролна информация или данни предназначени за протоколи от по-горните слоеве на OSI модела.
FCS (Frame Check Sequence) - съдържа стойността изчислена от CRC алгоритъма, служи за проверка дали кадъра е пострадал по време на транспорта.
ED (End Delimiter) - съдържа уникални символи определящи края на кадъра.

FS (Frame Status) - позволява на станцията изпращач да провери дали кадъра е бил разпознат и правилно копиран от станцията местоназначение.

Като допълнителни примери могат да се видят варианти за транслация на кадри между стандартите FDDI и Ethernet на ниво MAC, показани на фигура 71 [24].



фигура 71. *Преобразуване на кадри между двата стандарта*

6.12. Стандарт 802.12 (100 VG – Any LAN) – 100 Mbps

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=18**

Стандартът IEEE 802.12 [38][5] е съвместим с технологиите Ethernet и Token Ring. Може да се използва като пример за среда с различен метод на достъп от представените до момента - “приоритетен достъп по заявка” (demand priority).

6.12.1. Физически слой

Физическата топология е тип „звезда“, с възможност за преминаване в тип “дърво” (фигура 73а), като съществува ограничение за нивата на концентраторите до 5. Средата за предаване използва различни видове кабели:

- UTP кабел категория 3 – с максимална дължина между устройствата 100 метра;
- UTP кабел категория 5 – с максимална дължина между устройствата 200 метра;
- Оптичен кабел – с максимална дължина между устройствата 2 километра.

Максималният брой устройства е 1024 (250 препоръчително), а скоростта на предаване е 100 Mb/s.

Съпоставянето с OSI модела може да се онагледи с фигура 72а. Слоевете, които се отнасят към физическото ниво са: PMI (Physical Medium Independent sublayer), MII (Medium Independent Interface), PMD (Physical Medium Dependent sublayer) и MDI (Medium dependent interface) (фигура 72б).

OSI модел	Номер на слой	IEEE 802.12
Канален	2	LLC
		MAC
Физически	1	PMI
		MII
		PMD
		MDI

а)

MAC	MAC Frame			
PMI	Scrambler 0	Scrambler 1	Scrambler 2	Scrambler 3
	5B6B Encoder	5B6B Encoder	5B6B Encoder	5B6B Encoder
MII	Preamble, Start Frame, End Frame Delimiter			
PMD	Two-Level NRZ	Two-Level NRZ	Two-Level NRZ	Two-Level NRZ

	Encoder	Encoder	Encoder	Encoder
	Transmit Pair 1/2	Transmit Pair 3/6	Transmit Pair 4/5	Transmit Pair 7/8
MDI	Channel 0	Channel 1	Channel 2	Channel 3

б)

фигура 72. *IEEE 802.12 и OSI*

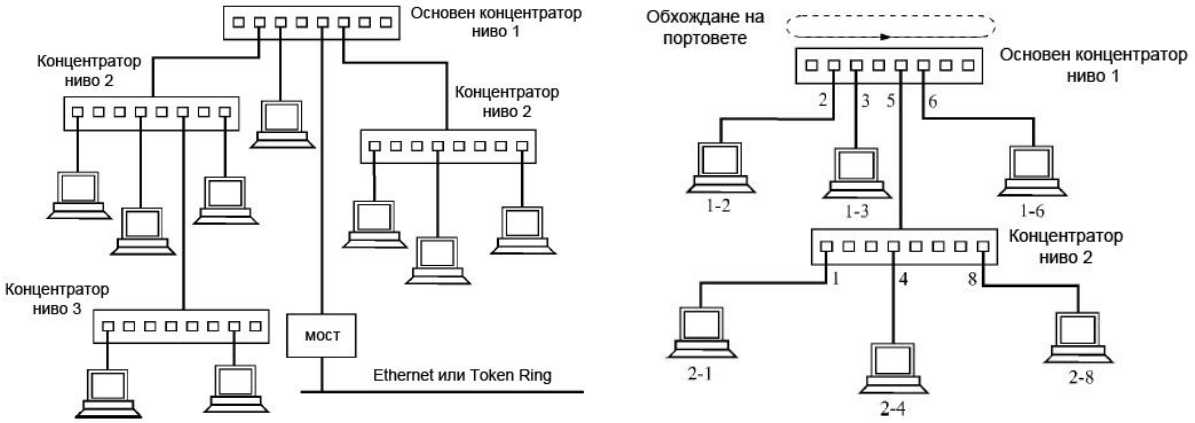
При използване на четири усукани двойки (UTP), като среда за пренос, данните се предават по четири канала т.е. MAC кадърът се разделя на части по 5 бита, които се предават по отделните канали. Целта е намаляване на потока данни по отделните проводници. Скремблирането означава случайно разбъркване на битовете в получените отрязъци, след което се използва кодиране 5B6B (5 бита се представят в 6 битова комбинация). MII слоя добавя встъпителна част (Preamble), начален (Start Frame Delimiter) и краен ограничител (End Frame Delimiter) към данните на всеки канал и ги предават за допълнително кодиране на PMD слоя. Той от своя страна се грижи за предаването по комуникационната линия. При използване на два канала за предаване (две усукани двойки или две оптични влакна) се използва мултиплексиране.

6.12.2. MAC-подслой

Използва протокола “приоритетен достъп по заявка” (demand priority) и се базира на предоставянето на концентратора на функции на арбитър, решаващ проблема с достъп до общата поделена мрежа. Използва се метод за разделяне на средата на две нива на приоритетност:

- нисък – на обикновените приложения;
- висок – за мултимедийните приложения, които са чувствителни на закъснение.

Необходимо е да се отбележи факта, че съвместимостта с технологиите Ethernet и Token Ring може да се осъществи само чрез мостове (фигура 73а), като се спазва условието, че всички концентратори, разположени в един логически сегмент, т.е. неразделени с мост, комутатор или маршрутизатор, трябва да бъдат конфигурирани да поддържат един и същи тип кадри. Ограничаването само до основния концентратор осигурява физическа топология тип „звезда“.



а) б)

фигура 73. *100VG-AnyLAN*

Цикличното обхождане на портовете на концентратора е представено на фигура 73б. Устройството, което предава подава заявка [37] към концентратора за предаване на кадър и определя приоритета му. Високоприоритетните приложения имат предимство за предаване пред останалите, като концентраторът следи да не се превишава определеното време за достъп до средата. Ако този тип заявки са много има опасност да се забави предаването на ниско-приоритетните. В този случай те получават висок приоритет на предаване.

Функционирането на структурата от фигура 73б може да се опише чрез следните примери:

- **Пример 1:** Ако всички компютри в мрежата едновременно подадат заявка за предаване с еднакъв приоритет последователността ще бъде следната: 1-2, 1-3, 2-1, 2-4, 2-8 и 1-6 (първата цифра от номерацията указва нивото на вложеност);
- **Пример 2:** Ако заявката от 2-8 е с по-висок приоритет ще има промяна в последователността: **2-8**, 1-2, 1-3, 2-1, 2-4 и 1-6.

6.13. IEEE 802.11 (Wi-Fi)

Адрес: **http://kmk.uni-plovdiv.org/kmk-lectures/mod/page/view.php?id=23**

IEEE 802.11 дефинира набор от стандарти за Wireless LAN/WLAN, разработени от работна група 11 на IEEE LAN/MAN Standards Committee (IEEE 802). Безжичните мрежи имат фундаментални характеристики, които ги правят значително по-различни от традиционните кабелни локални мрежи. Някои държави налагат специфични изисквания за радио

оборудването в допълнение към тези определени в стандарта 802.11. При кабелните локални мрежи, MAC адреса отговаря на физическото местоположение. Това е безусловно прието в проектирането на кабелни локални мрежи. В стандарта IEEE 802.11, адресируемата единица е станцията (STA). Станцията е приемника на съобщенията, но не е физически фиксирана на определено място.

6.13.1. Физически слой

Физическите слоеве (PHY) използвани в IEEE 802.11 стандарта поначало са различни от тези използвани с кабелни среди. За PHY протоколите от стандарта IEEE 802.11 може да се отбележи, че:

- използват преносна среда, която не е с ясно дефинирани граници и е по-ненадеждна от кабелната среда;
- използват динамични топологии;
- нямат пълна свързаност (станциите могат да бъдат скрити една от друга);
- имат характеристики, които са вариращи във времето.

Поради ограничението на безжичния PHY (физически слой) обхват необходимостта от покриване на определени географски разстояния налага изграждането на отделни сектори с ограничено покритие. Едно от изискванията на стандарта IEEE 802.11 е да се справя както с мобилни, така и с преносими станции. Преносима станция е тази, която може да сменя своето местоположение, но се използва само на фиксирана позиция. Мобилните станции имат достъп до локалната мрежа по време на движение. Характерно за мобилните станции е, че те се захранват предимно с батерии. Затова управлението на захранването е важен фактор. Например, не може да бъде прието, че приемникът винаги ще е включен. Функционалността на стандарта IEEE 802.11 трябва да управлява мобилността на станцията в MAC подслоя.

фигура 74 *Устройства за безжична комуникация*

6.13.2. Технологии за пренос

6.13.2.1. С широк радиоспектър

- *скачаща честота* (FHSS) – разделя честотната лента на подканали. В даден момент използва само един канал. Сигналът скача според предварително уговорен ред и честота;
- *директна поредица* (DSSS)– използва различните подканали в пореден ред;
- *ортогонална честота* (OFDM) – радиосигнала се разделя на множество подсигнали и едновременно се излъчва на съвсем леко различаваща се честота в общия канал.

6.13.2.2. С тесен или еднолентов радиоспектър

Използват само един канал (микровълнов обхват). Покривно разстояние до 42м на открито и 12м на закрито. Скорост на предаване - 15 Mbps.

6.13.2.3. Инфрачервени

- Директна (разпространява се в една посока);
- Дифузна (разпръсква във всички посоки).

Покривно разстояние до 30 м.

6.13.2.4. Лазерни

Концентриран лазерен лъч, разпространяващ се в една посока. Покрива големи разстояния.

6.13.3. Разновидности на стандарта

В таблица 3 са представени разновидностите на стандарта и техните основни параметри.

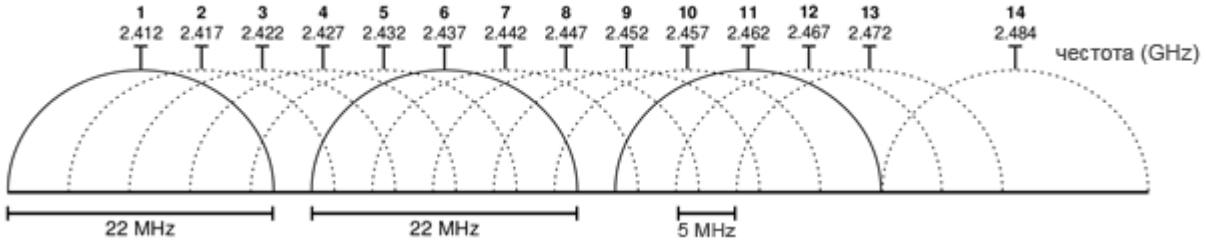
Под-стандарт	Дата на изд.	Оперативна честота	Data Rate (Typ)	Data Rate (Max)	Обхват (в сграда) в метри	Обхват (на открито) в метри	Модулация
Legacy	1997	2.4 GHz	0,9 Mbit/s	2 Mbit/s	~20	~100	DSSS, FHSS
802.11a	1999	5 GHz	23 Mbit/s	54 Mbit/s	~35	~120	OFDM
802.11b	1999	2.4 GHz	4.3 Mbit/s	11 Mbit/s	~38	~140	DSSS
802.11g	2003	2.4 GHz	19 Mbit/s	54 Mbit/s	~38	~140	OFDM, DSSS
802.11n	2009	2.4 GHz 5 GHz	130 Mbit/s	300 Mbit/s	~70	~250	OFDM

802.11y	2008	3.7 GHz	23 Mbit/s	54 Mbit/s	~50	~5000	OFDM
802.11ac	2013	5 GHz	1.3 Gbit/s	2.33 Gbit/s	~35	~465	OFDM

таблица 3 *Спецификации на стандарта IEEE 802.11*

При разновидностите на стандарта 802.11b, 802.11g, и 802.11n в 2.4 GHz обхват теоретично се предоставят до 14 канала (2.400–2.500 GHz) с отместване от 5 MHz и ширина от 22 MHz (фигура 75). Като втора възможност за 802.11n е използването на ширина от 40 MHz за по-висока пропускателна способност за сметка на броя на каналите.

На практика за избягване на смущения, могат да се използват само три или четири канала. Такъв тип комбинации са: (1, 6, 11), (2, 7, 12), (3, 8, 13), (4, 9, 14 - ако е позволен) или (5, 10, 14 - ако е позволен). При канал с ширина 40 MHz това са 3 и 11.



фигура 75 *Разпределение на каналите за безжична комуникация при 2.4 Ghz*

Най-новият стандарт 802.11ac оперира в 5GHz обхват като добавя ширина на каналите от 80MHz и 160MHz.

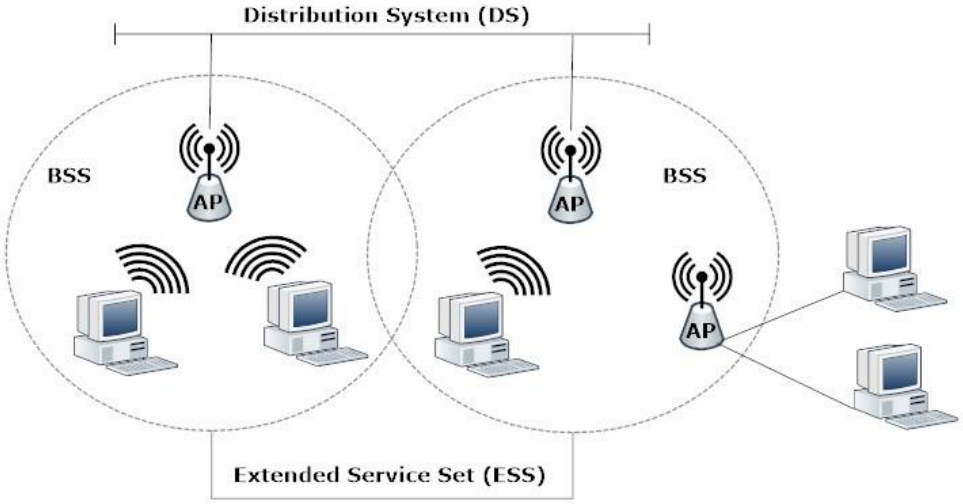
6.13.4. Логическа архитектура

Архитектурата на IEEE 802.11 се състои от няколко компонента, които си взаимодействат, за да предоставят безжична свързаност. Тези компоненти могат да поддържат мобилността на станцията, която е прозрачна за по-горните слоеве.

6.13.4.1. Основен набор от услуги (BSS - Basic Service Set)

Основният набор услуги (BSS) е изграждащ блок на IEEE 802.11 локална мрежа (фигура 76). Формира се при комуникация на две или повече станции. BSS покрива единична радио-честотна област или клетка. Когато станцията се отдалечава от точката за достъп, скоростта на обмен на информацията се понижава. При излизане от обхвата на своя BSS се прекъсва комуникацията с другите членове на набора от услуги.

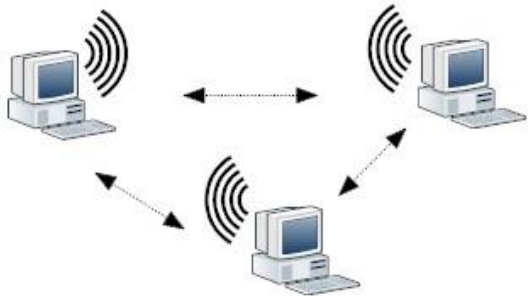
BSS използва инфраструктурен режим, който се нуждае от точка за достъп. Всички станции комуникират чрез точката за достъп, а не директно. Всеки BSS има само един SSID (идентификационен номер на набора от услуги).



фигура 76 *Архитектура на стандарта IEEE 802.11*

6.13.4.2. Независим BSS (IBSS- Independent Basic Service Set)

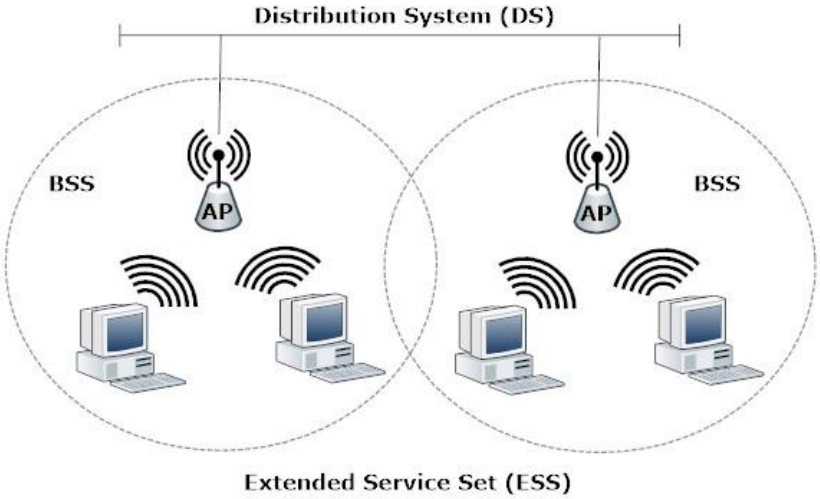
В този режим на работа IEEE 802.11 станциите комуникират директно. Заради този тип услуга, локалната IEEE 802.11 мрежа често е формирана без предварително планиране и е наричана Ad-Hoc мрежа. Понеже IBSS (фигура 77) се състои от станции, които са директно свързани, той се нарича още „peer-to-peer“ мрежа. По дефиниция има само един BSS и няма дистрибуционна система (Distribution System – DS). IBSS може да има произволен брой членове. За комуникация извън IBSS, една от станциите трябва да работи като шлюз или маршрутизатор.



фигура 77 *Independent Basic Service Set*

6.13.4.3. Дистрибуционна система (DS- Distribution System)

Физическите ограничения са причината за формирането на разширен набор от услуги (extended service set – ESS). ESS е изграден от множество BSS реализирани чрез точки за достъп (фигура 78). Точките за достъп са свързани към обща дистрибуционна система. Дистрибуционната система може да бъде кабелна, безжична, локална мрежа или WAN. Безжичната архитектура на IEEE 802.11 е специфицирана, независимо от физическите характеристики на дистрибуционната система. Дистрибуционната система позволява поддръжка на мобилни устройства чрез предоставяне на услугите, нужни за управлението на процеса по обвързване с адресите (address mapping) и интеграцията на множество BSS. Информацията се движи между BSS и дистрибуционната система чрез точка за достъп.



фигура 78 *Дистрибуционна система (DS)*

6.13.4.4. Разширен набор от услуги (ESS)

Разширения набор от услуги (ESS) е определен като две или повече BSS свързани чрез обща дистрибуционна система (фигура 78). Това позволява създаването на безжична мрежа с произволен размер и сложност. Както при BSS, всички пакети в ESS трябва да преминават през една от точките за достъп. Основната концепция е, че на ниво LLC слой ESS мрежата изглежда като IBSS или единична BSS мрежа. Станции в ESS могат да комуникират директно, а мобилните станции могат да се движат от една BSS в друга в същата ESS, прозрачно спрямо LLC слоя.

6.13.4.5. Поддържани услуги от стандарта

Поддържаните услуги се групират в две основни направления - услуги на дистрибуционната система (Distribution System Services-DS) и услуги на станцията (Station Services).

Услугите на дистрибуционната система са:

1. Асоциация (Association)

Присъединяването на станция към BSS инфраструктура за използване на LAN се осъществява чрез AP. Извършва се процеса *асоциация*. Асоциациите са динамични по естество. Станцията се асоциира с едно AP, което позволява на DS да я локализира еднозначно. Асоциацията поддържа *No-transition* модела, при който станцията не се движи или го прави в обхвата на собствения BSS. Останалите типове мобилност са:

- *BSS- transition* – когато станцията се движи между няколко BSS с общ ESS;
- *ESS-transition* – когато станцията прави преход между BSS от различни ESS.

2. Реасоциация (Reassociation) - позволява смяна на асоциирането на станцията от едно към друго AP.

Асоциацията и реасоциацията се инициализират от станцията.

3. Дисасоциация (Disassociation) - процес на прекъсване на асоциацията на станцията с AP. В този случай станцията не може да приема и предава.

Дисасоциацията може да се предизвика от всяка от страните.

4. Дистрибуция (Distribution) - процес на пренасяне на данните от изпращача до получателя. Съобщението преминава от AP на изпращача през DS системата до AP на получателя. Ако участниците на комуникацията са в една BSS, то AP е едно и също.

5. Интеграция (Integration) - изходното AP е портал към друг тип IEEE 802.x локални мрежи.

Първите три групи услуги са предназначени за справяне с мобилността на станциите.

Услугите на станциите са:

1. Автентикация – извършва се между две станции при IBSS или станция и AP при BSS. Съществуват два вида услуги по удостоверяване при IEEE 802.11:

- Open System Authentication (OSA) – всеки получава свързване;
- Shared Key Authentication (SKA) – чрез използване на шифриращи алгоритми.

2. Деавтентикация - когато станцията или AP прекрати автентикацията. В този случай станцията е *дисасоциирана*.

3. Поверителност – позволява използването на шифриращ алгоритъм. При комуникацията може да се използва или не такъв, което означава, че трафикът може да бъде шифриран или не.

4. MSDU (Medium Service Data Unit) доставка - гарантира успешната размяна на кадри между точките за достъп.

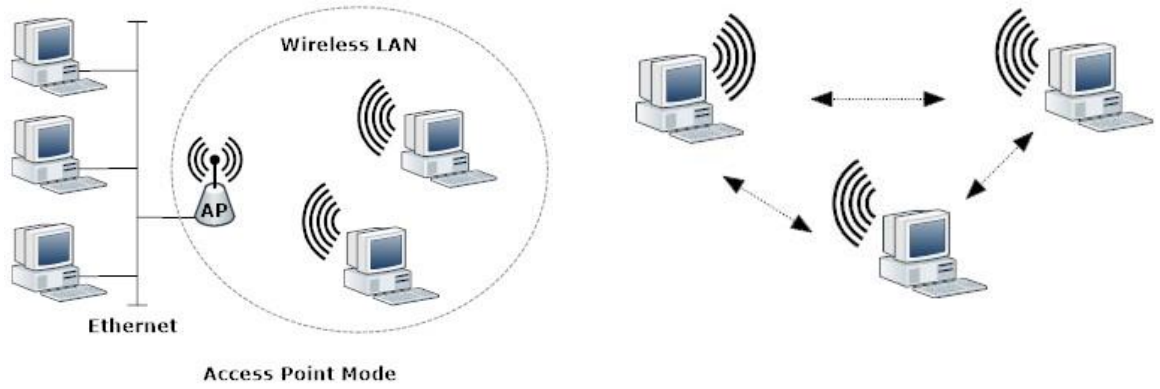
6.13.4.6. Роуминг

Роумингът е процес или способност на безжичен клиент да се премества от една клетка или BSS към друга, без да губи свързаността към мрежата. Точките за достъп си предават взаимно клиента, като целия процес е невидим за клиента. Стандарта IEEE 802.11 не определя как роуминга ще се реализира, но определя основните изграждащи блокове, които включват активно и пасивно сканиране и реасоциационен процес. Реасоциацията с точката за достъп трябва да възникне, когато станция се движи от една точка за достъп към друга

6.13.5. Типове безжични мрежи

Wireless мрежите имат два различни режима на работа:

- *infrastructure* - представлява *WLAN* и *LAN* мрежа, комуникиращи една с друга чрез *Access Point* (фигура 79а);
- *Ad-Hoc* - конфигурация от компютри, оборудвани с *Wireless* устройства, комуникиращи директно един с друг (фигура 79б).

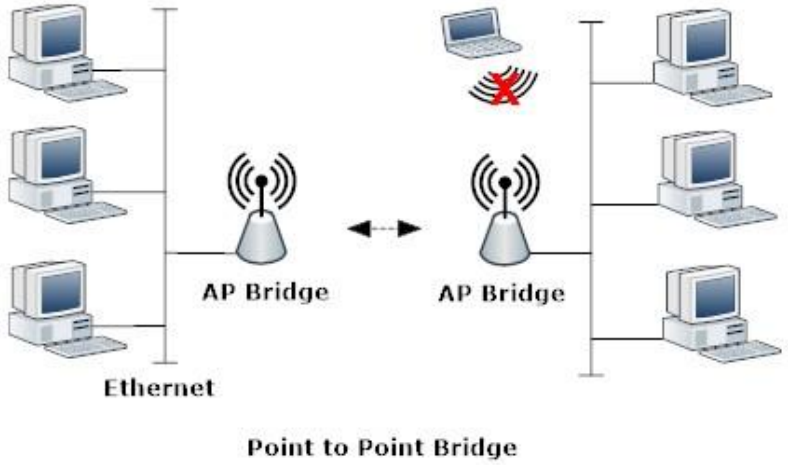


а) *infrastructure* б) *Ad-Hoc*

фигура 79 *Режими на работа*

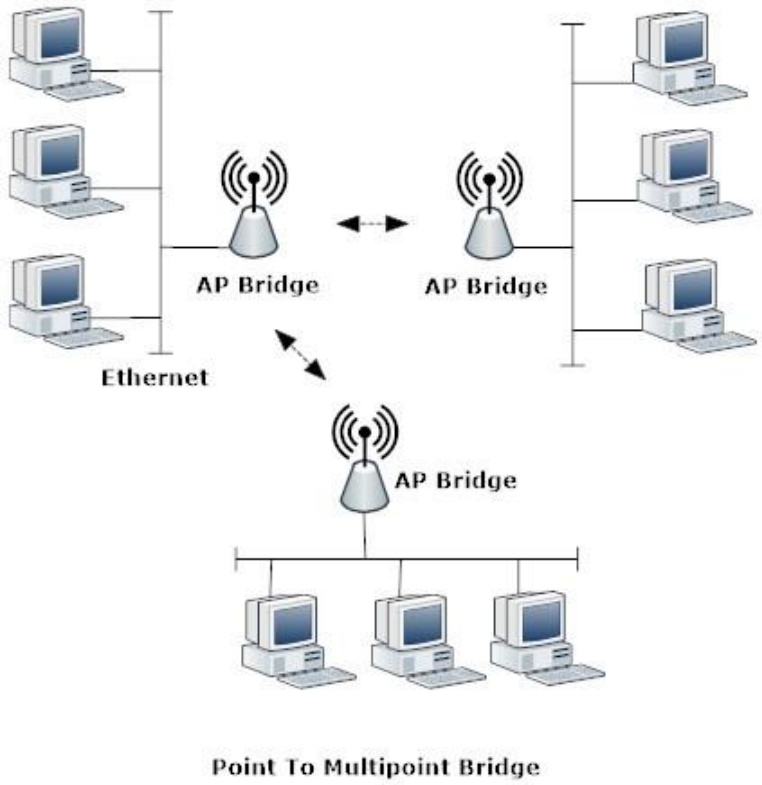
6.13.6. Режими на работа на Access Point устройствата

- *Access Point* (*AP*) – този режим позволява към устройството да се свързват крайни безжични клиенти или *AP* устройства в клиентски режим, предоставящи свързаност на един или няколко компютъра (фигура 79а);
- *AP Client* – режимът позволява на *AP* устройство да работи като клиент на друго *AP* устройство и да предоставя безжична свързаност на локална мрежа. Налични са всички услуги на безжичната мрежа, включително автентикация на клиенти и т. н.;
- *Point To Point Bridge (P2P Bridge)* – използва се за безжично свързване само на две устройства (фигура 80);



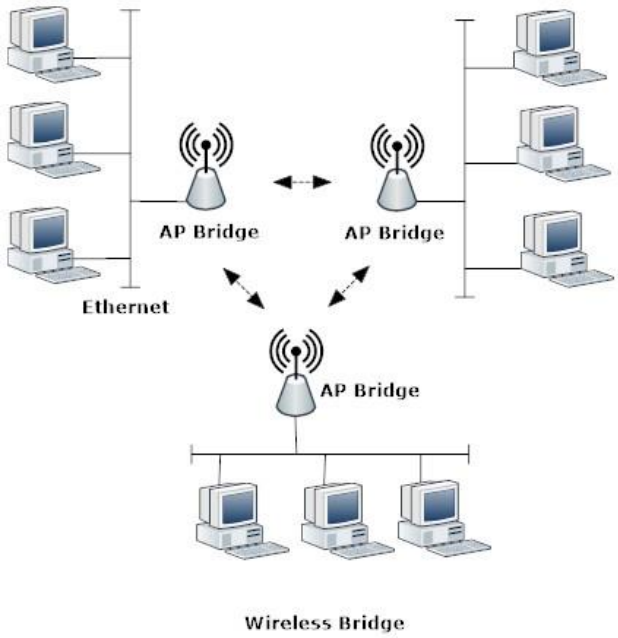
фигура 80 *Point To Point Bridge*

- *Point To Multipoint Bridge (P2MP Bridge)* – режимът позволява към централно безжично устройство да се свържат други такива в бридж режим (фигура 81);



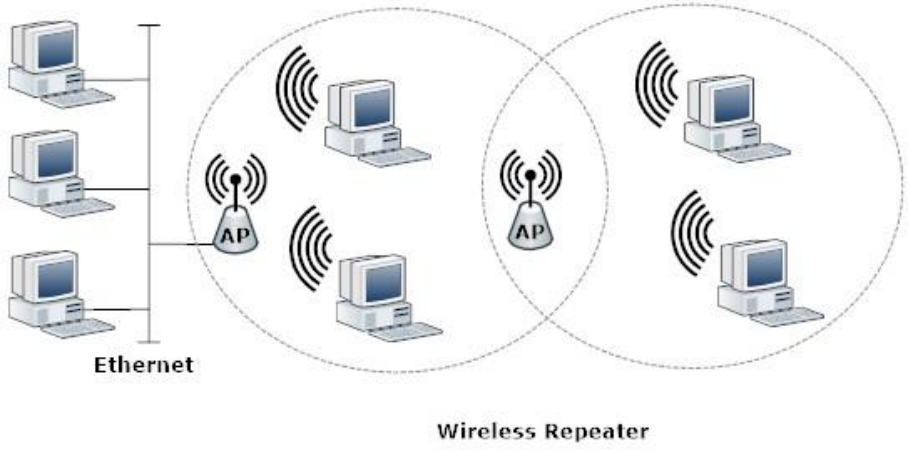
фигура 81 *Point To Multipoint Bridge*

- *Ad-Hoc Bridge* – при тази конфигурация АР устройствата работят като бриджове в Ad-Hoc режим (фигура 82). Мрежата е аналогична на горната с разликата, че не съществува едно централно устройство, а всеки бридж може да се свързва с другите;

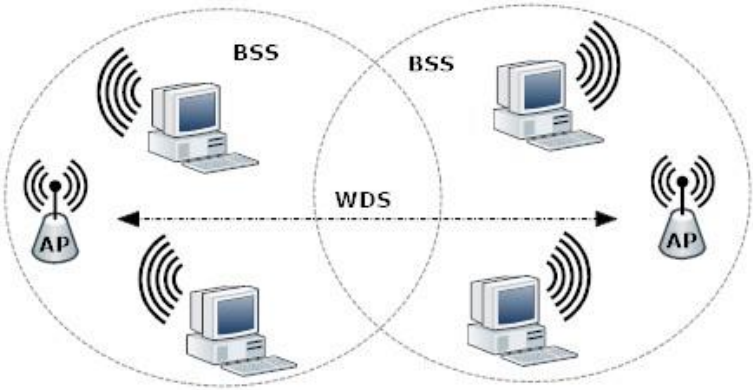


фигура 82 *Ad-Hoc Bridge*

- *Wireless Repeater* или *WDS* (*Wireless Distribution System*) – режимът позволява на едно *AP* устройство да разширява областта на покритие на друго *AP* (фигура 83).



фигура 83 *Wireless Repeater*



фигура 84 *WDS (Wireless Distribution System)*

6.13.7. IEEE 802.11 MAC слой

Отнасянето към OSI модела може да бъде представено чрез фигура 85, където се вижда възможността на MAC подслоя да комуникира с различните режими на предаване на физическо ниво.

OSI модел	Номер на слой	IEEE 802.11			
Канален (MAC)	2	LLC			
		IEEE 802.11MAC			
Физически (PHY)	1	IR	FHSS	DSSS	OFDM

фигура 85. *IEEE 802.11 и OSI*

Използването на радиовълни налага разделянето на физическия слой на два подслоя:

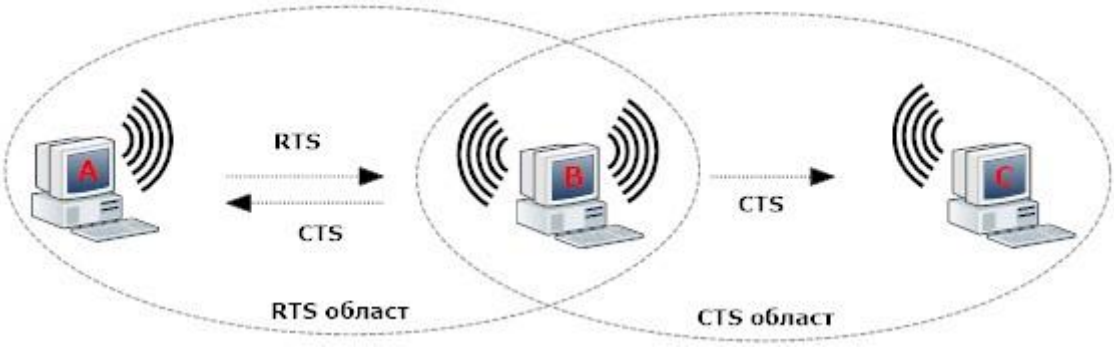
- PLCP (Physical Layer Convergence Protocol) подслой – комуникира директно с MAC слоя чрез команди (примитиви) през точка за достъп на услуги (SAP). PLCP подготвя MAC данните (MPDUs) за предаване към PMD подслоя като добавя PHY специфичен преамбюл и заглавно полета към MPDU, които съдържат информация, необходима на предавателите и приемниците от физическия слой. Получената нова протоколна единица се нарича PPDU (PLCP protocol data unit). PLCP също така доставя входящите рамки от безжичната среда на MAC слоя.
- PMD (Physical Medium Dependent sublayer) подслой – осигурява предаване и приемане на данните на ниво физически слой между две станции по безжичната среда под ръководството на PLCP подслоя. Двата подслоя си комуникират чрез примитиви през SAP.

Основният метод на достъп, който се използва е множествен достъп с разпознаване на носещата и избягване на колизии (CSMA/CA- Carrier sense multiple access with collision avoidance). Този метод се опитва да избегне колизиите преди да започне истинското предаване на данни. Безжичният адаптер прослушва споделената среда. Ако сигналът съдържа данни, станцията трябва да изчака. При свободна среда може да предава. Този подход се стреми да осигури и справедлив достъп до средата за предаване. За целта се използват две времеви техники:

- минимално времезакъснение – гарантира, че една станция няма да заеме цялата честотна лента. След всяко успешно предаване на кадър предаващата станция трябва да изчака определеното минимално времезакъснение, преди следващия трансфер;
- случаен интервал при колизия – станцията изчаква произволен интервал преди отново да започне да прослушва канала. Ако каналът е чист може да започне предаване на следващия кадър. При зает канал станцията изчаква още един такъв интервал и т.н.

При предаване може да бъде използвана опцията RTS/CTS (Request to Send/Clear to Send). При този режим станцията-подател изпраща RTS заявка до получателя и очаква от него CTS отговор, което е знак, че може да предава, а другите станции, попаднали в двете области, трябва да изчакат (фигура 86). RTS/CTS не е задължителна за използване, но спомага за намаляването на конфликтите, особено при наличието на “скрити” станции. Този проблем възниква, когато станциите A и C предават

едновременно към B, но не могат да се чуят поради ограничението на обхвата им (фигура 86).



фигура 86 *"Скрити" станции*

Получаването на някои кадри изисква приемащата станция да отговори с потвърждение, в повечето случаи с ACK пакет, ако FCS (Frame Check Sequence) на получения кадър е верен. Липсата на потвърждение показва на предавателя, че е възникнала грешка и трябва да предаде кадъра отново. Възможно е приемника да е приел кадъра правилно, а грешката да е в доставянето на потвърждението. За инициатора на връзката, тези две състояния са неразличими.

Подходът, които реализира описания по-горе метод на съперничество се нарича Дистрибутивна Координатна Функция (Distributed Coordination Function – DCF). DCF се прилага във всички станции, независимо дали работят в ad-hoc или инфраструктурен режим.

IEEE 802.11 MAC може да включва също и опционен метод за достъп, наречен PCF – Point Coordination Function, който създава достъп без съперничество (contention-free - CF). Методът PCF може да бъде използван само при инфраструктурен режим.

Методите DCF и PCF могат да работят конкурентно в един и същ BSS. Когато това се случва, двата метода се редуват. Всеки CF период е следван от период на съперничество. В допълнение, всички предавания по метода PCF могат да използват междукадрово разстояние (IFS), което е по-малко от това използвано за кадри предавани по начина на DCF. Употребата на по-малко междукадрово разстояние означава, че точково координирания трафик ще има приоритетен достъп до средата над станциите работещи в DCF режим.

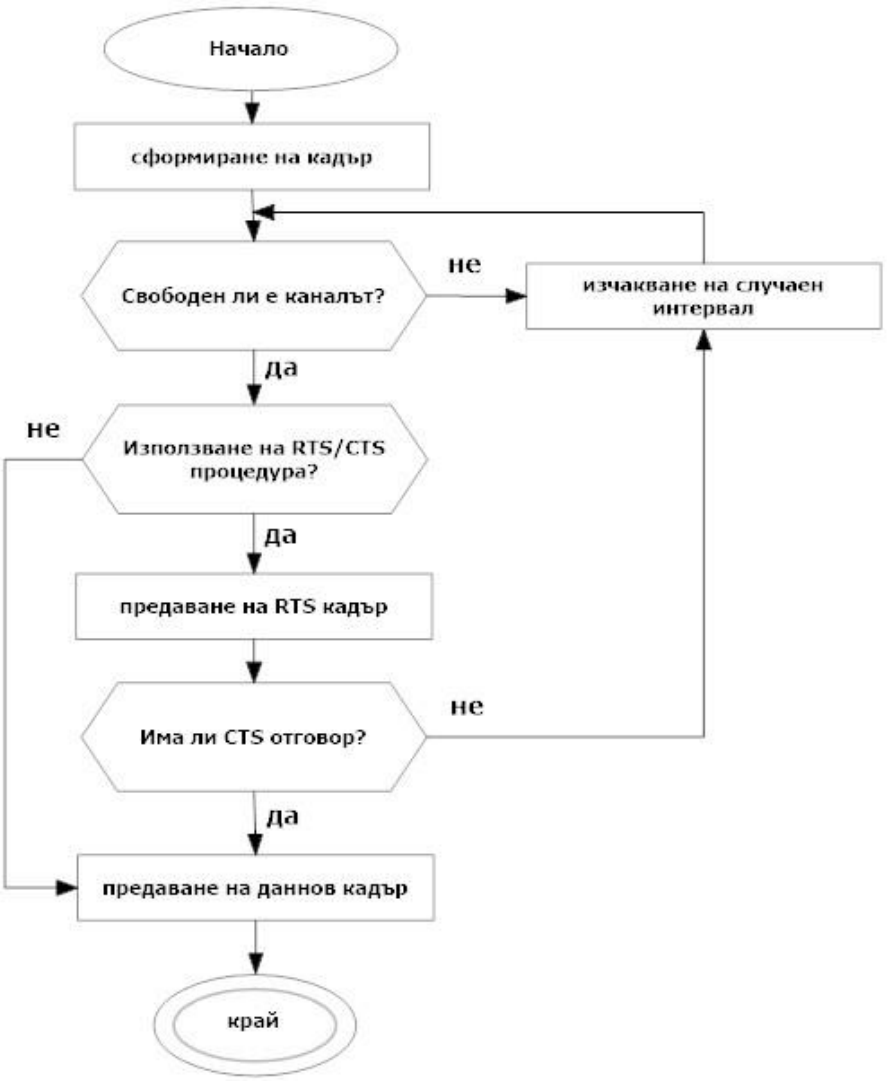
Времевият интервал между кадрите е наречен междукадрово разстояние (InterFrame Space – IFS). Всеки IFS интервал се определя като времето от последния бит на предния кадър до първия бит на следващия кадър. Дефинират се четири различни вида междукадрови разстояния, които предоставят приоритетни нива на достъп до безжичната среда:

- SIFS (short interframe space) е кратко междукадрово разстояние. Използва се за предаване с висок приоритет (например, RTS/CTS и положителни потвърждения);
- PIFS (PCF interframe space) е PCF междукадрово разстояние;
- DIFS (DCF interframe space) е DCF минимално изчакващо междукадрово разстояние;
- EIFS (Extended interframe space) е разширено междукадрово разстояние без фиксирана дължина. Използва се при възникване на грешки при предаване на кадъра.

Различните междукадрови разстояния са независими от скоростта на станцията. Междукадровите времена се дефинират като времена без предаване и зависят от физическия слой.

Физическите механизми за тестване на средата за множествен достъп се предоставят чрез физическия слой. MAC слоят предоставя виртуален механизъм за тестване на средата за множествен достъп. Този механизъм е известен като вектор на мрежовото разпределение (network allocation vector - NAV). NAV поддържа предсказване на бъдещия трафик предаван по средата, на базата на информация от полето за продължителност (*Duration*) от заглавната част на уникаст кадрите. То определя времето в микросекунди, през което ще бъде заета средата за предаване на рамката. При прослушване станциите проверяват това поле и инициализират своя NAV вектор. Той определя продължителността на изчакване на станцията за достъп до средата.

Следващата блок-схема пресъздава алгоритъма на CSMA/CA за предаване на кадър.



фигура 87 *Предаване на кадър*

Трите основни вида кадри на стандарта са:

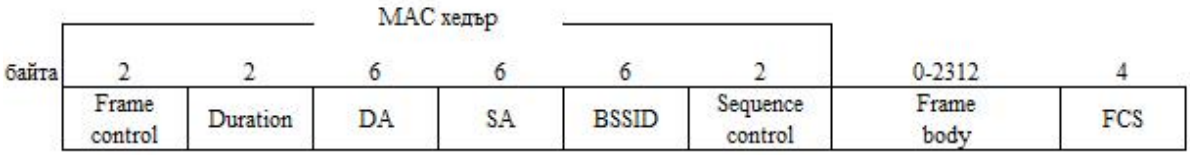
- информационни - използват се за предаване на информация;
- кадри за контрол (контролират достъпа до средата) - RTS (Request To Send - заявка за изпращане, която подателят изпраща на получателя преди да започне предаването на данните), CTS (Clear To Send) и ACK (Acknowledgment – потвърждение, с което получателят информира подателя за успешното получаване на данните);
- кадри за управление - като информационните кадри, но обменят управляваща информация и не се предават към по-горен слой.

Управляващите кадри се разделят на няколко вида:

- Association Request frame (заявка за асоциация) – мрежовият адаптер стартира този процес чрез Association Request кадър, носещ информация за адаптера (напр. поддържани скорости на предаване) и SSID на мрежата, към която желае да се включи.
- Association Response frame (отговор на заявката за асоциация) – AP устройството изпраща този кадър на мрежовият адаптер, поискал асоциация с него, в който се съдържа приемане или отказ за асоциация. Ако AP устройството приема мрежовият адаптер, кадърът съдържа и служебна информация за самата асоциация, напр. ID на асоциацията и поддържани скорости на предаване от страна на AP устройството.
- Reassociation Request frame (заявка за реасоциация) – ако безжичен адаптер, асоцииран с дадено AP устройство попадне в обсега на друго AP устройство от същата мрежа, имащо по-силен сигнал, мрежовият адаптер изпраща такъв кадър на новото AP устройство за нова асоциация с него.
- Reassociation Response frame (отговор на заявката за реасоциация) – AP устройството изпраща този кадър, съдържащ приемане или отказ на адаптера, поискал реасоциацията. Както при Association Request кадъра, ако AP устройството приеме реасоциацията, изпраща ID на асоциацията и поддържани скорости на предаване.
- Disassociation frame (кадър за дисасоциация) – кадърът се изпраща при заявка за дисасоциация. Например, при изключване на безжичен мрежов адаптер се излъчва този тип кадър, за уведомяване на AP устройството към което е асоцииран. Действието от негова страна е свързано с премахване на всички данни за този мрежов адаптер от таблицата на асоциациите.
- Authentication frame (автентикационен кадър) – кадърът стартира процеса на идентификация на мрежовия адаптер при AP устройството.
- Deauthentication frame (деавтентикационен кадър) – изпраща се при желание за прекратяване на защитена комуникация.
- Beacon frame – AP устройствата периодично изпращат такива кадри, за да оповестят своето присъствие. Съдържат SSID-то на мрежата и друга информация, полезна за мрежовите адаптери попаднали в обхвата им.

- Probe Request frame – изпращат се от станциите при сканиране за AP устройства или BSS.
- Probe Response frame – изпраща се от мрежата в отговор на Probe Request кадъра и съдържа информация за съвместимостта на устройството, подало заявката, с другите станции, поддържани скорости на предаване и др.

Структурата на управляващ кадър е показана на фигура 88.



фигура 88 *Структура на управляващ кадър*

MAC хедърът е общ за всички кадри от този тип.
BSSID – безжичните станции проверяват това поле и обработват само тези кадри, чийто източник е AP-то, към което са асоциирани. Изключение правят Beacon кадрите.
Duration – при DCF се използва за блокиране на достъпа до средата при предаване на кадър.
Frame body - съдържа полета с фиксирана дължина (fixed fields) и полета с променлива дължина (information elements).

Към фиксираните полета, принадлежащи на частта Frame body се причисляват:

- поле за идентифициране на използвания алгоритъм за автентикация (Authentication Algorithm Number) – двубайтово поле със стойности показани в таблица 4.

Стойност	Предназначение
0	Open System authentication
1	Shared Key authentication
2-65535	Reserved

таблица 4 *Стойности*

- поле за означаване на последователността на кадрите при транзакция при процеса на автентикация (Authentication Transaction Sequence Number) – двубайтово поле, заемащо стойности в интервала [1;65535];
- поле за определяне на времевите единици (time units) между предаването на Beacon кадрите (Beacon Interval) - двубайтово поле,

определящо броя на времевите единици. Една такава единица се отбелязва още с означението TU и се равнява на 1024 microseconds (μs). Стандартният брой TU е 100;

- поле за рекламиране на възможностите на мрежата (Capability Information) – двубайтово поле, използвано основно в Beacon трансмисиите. Станциите, обработващи тези кадри, анализират отделните битове и установяват дали могат да удовлетворят изискванията на мрежата за евентуално присъединяване към нея.
- Поле за идентификация на текущото AP (Current AP Address) – шест байтово число, указващо MAC адреса на AP устройството, към което е асоциирана станцията. Използва се за улесняване на асоциацията и реасоциацията.
- Поле за определяне на броя на Beacon интервалите, които станцията трябва да изчака преди да започне прослушване за Beacon кадри (Listen interval) – двубайтово поле, показващо колко дълго точките за достъп могат да пазят буферираните кадри, докато станциите спят. Всяка станция, за да запази батерията си по-дълго време, изключва безжичния си интерфейс за определения от това поле период от време, през което AP устройството буферира кадрите предназначени за нея.
- Поле за идентификация на асоциациите (Association ID) – двубайтово поле, съдържащо идентификационен номер на активната асоциация.
- Поле за синхронизация (Timestamp) – осем байтово поле, позволяващо синхронизация между станциите в една BSS.
- Поле за идентифициране на причината за некоректен изпращач (Reason Code) – двубайтово поле, съдържащо определени кодове дефинирани за некоректни действия от страна на изпращача.
- Поле за указване на статуса на операцията (Status code) – двубайтово поле, съдържащо стандартизиращ код, указващ успешно или неуспешно завършена операция.

Информационните елементи, принадлежащи на частта Frame body са компоненти с променлива дължина, обхващащи три полета: ID номер; предварително дефинирано поле за дължина и променлива компонента. Заеманите стойности на ID и значението на информационните елементи са показани в таблица 5.

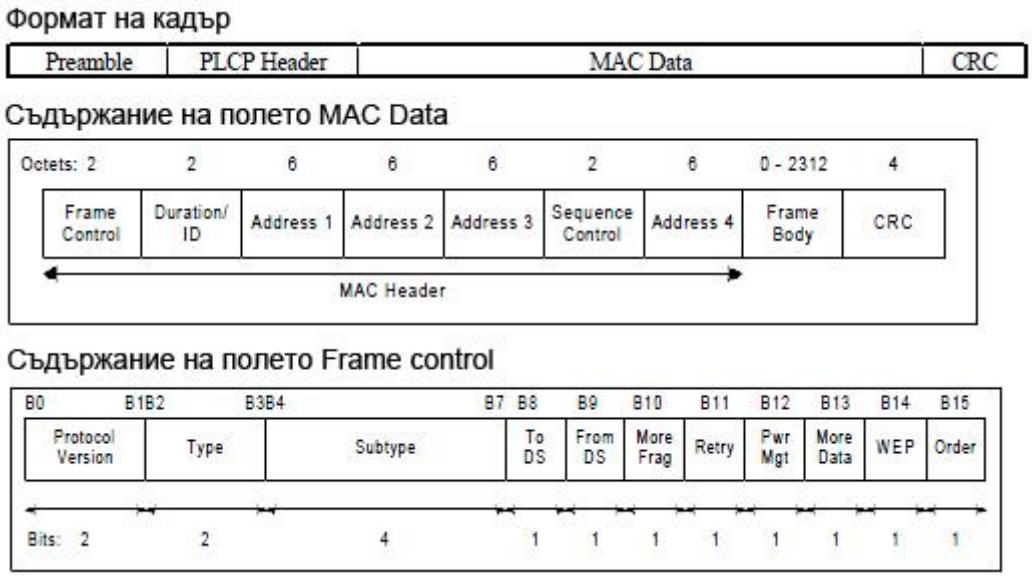
ID	Означение	Предназначение
0	Service Set Identity (SSID)	Предоставя името на мрежата
1	Supported Rates	Указва скоростите, поддържани от безжичната мрежа
2	FH Parameter Set	Съдържа необходимите параметри за мрежи използващи скачаща честота (frequency-hopping 802.11 network)
	Параметри	
	Dwell Time	Двубайтово поле, съдържащо продължителността, за която се използва всеки подканал
	Hop Set	Еднобайтово поле, определящо съвкупността от скащачи модели, който се използват. За стандарта IEEE 802.11 са дефинирани няколко такива модела.
	Hop Pattern	Еднобайтово поле, идентифициращо използвания модел
	Hop Index	Еднобайтово поле, съдържащо указател към активната, в момента, точка от изпълняваната последователност. Всеки модел поддържа определена последователност при скачане между подканалите.
3	DS Parameter Set	Еднобайтово поле, указващо текущия канал в последователността за мрежи от тип Direct-sequence 802.11 networks.
4	CF Parameter Set	Изпраща се от AP, поддържащо contention-free операция.
5	Traffic Indication Map (TIM)	AP устройството буферира кадрите за станциите преминали в low-power режим. Периодично то се опитва да достави буферираните кадри за станциите преминали в този режим. Това поле показва кои от тях имат буфериран трафик, очакващ тяхното събуждане.
6	IBSS Parameter Se	Полето се използва единствено при IBSS Beacon кадрите и съдържа продължителността в TU (time unit) между ATIM (announcement traffic indication map) кадри в IBSS.
16	Challenge text	Shared-key authentication системата на 802.11 изисква станцията успешно да дешифрира и дешифрира данните изпратени в това поле.

таблица 5 *Информационни елементи*

Общ формат на кадъра за данни е представен на фигура 89. Разграничават се следните основни компоненти:

- MAC заглавна част (MAC Header) – състои се от полета за контрол на кадри, продължителност, адреси и контролна информация за последователността.
- Тяло на кадъра с променлива дължина (Frame Body) - съдържа информация характерна за неговия тип. Например, в информационните кадри, това включва информация от по-горен слой.

- Проверка на кадъра (FCS) - 32-bit-ова проверка (CRC).



фигура 89. *Формат на даннов кадър*

Наличието на някои полета зависи от типа на кадъра за данни. Различните типове кадри се категоризират в зависимост от функциите, които изпълняват. Например, полето Frame Control съдържа контролни битове като някои от тях указват начина на интерпретация на полетата на кадъра. В таблица 6 са показани значенията на адресите в зависимост от състоянието на контролните битове ToDS и FromDS.

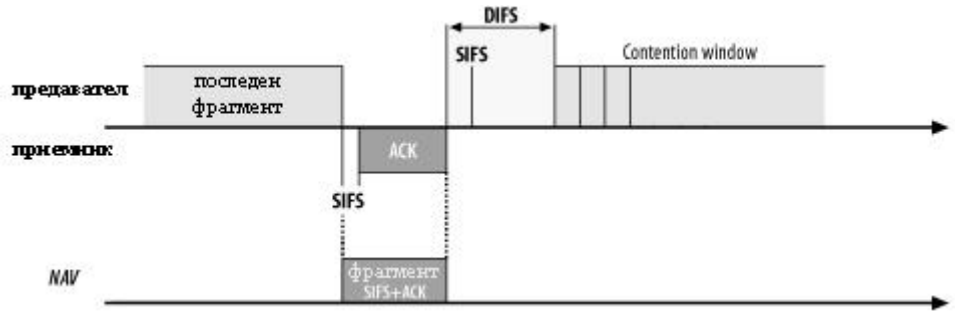
Трансмисия	ToDS	FromDS	Address 1 (приемник)	Address 2 (предавател)	Address 3	Address 4
IBSS	0	0	DA	SA	BSSID	не се използва
към AP (инфр.режим)	1	0	BSSID	SA	DA	не се използва
от AP (инфр.режим)	0	1	DA	BSSID	SA	не се използва
WDS	1	1	RA	TA	DA	SA

таблица 6 *Значение на адресните полета според контролните битове*

Всяка BSS се идентифицира с 48 битов идентификатор. При инфраструктурен режим това е MAC адреса на безжичния интерфейс на AP устройството. При IBSS се използва случаен генератор.

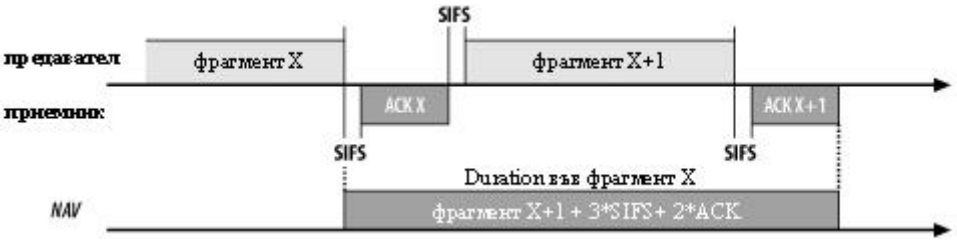
Полето Duration пренася стойността за NAV, който забранява достъпа до средата за време определено от тази стойност. На фигура 90 е представено предаването на финален фрагмент за кадър. Този фрагмент е необходимо да запази време за достъп до средата за собственото си

потвърждение като за целта в полето Duration заложи времето за предаване на SIFS+ACK.



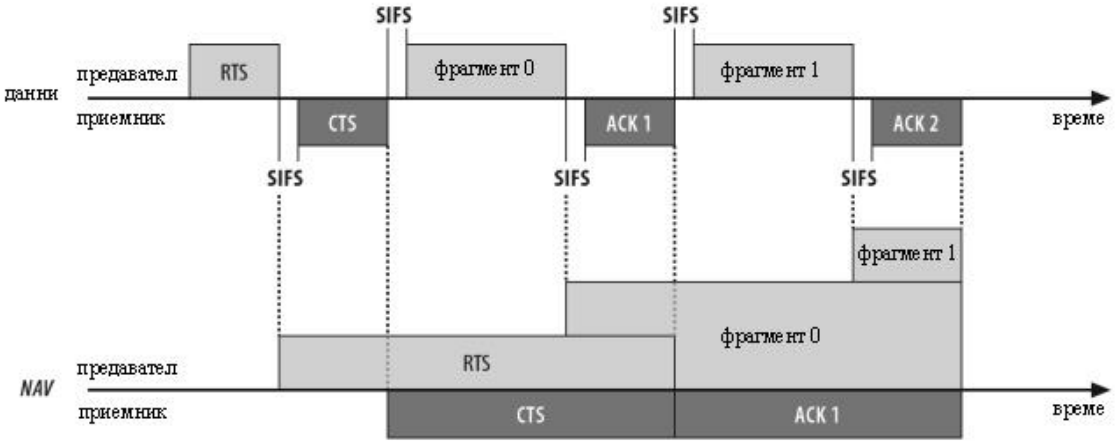
фигура 90 *Полето Duration за последен фрагмент от кадъра*

След приключване на предаването на кадъра следва DIFS интервал и евентуално период наречен contention window (още backoff window). Разделен е на слотове. Дължината на слота се определя от средата за предаване. Изборът на слот от страна на станцията е случаен. При повече кандидати за предаване станцията избрала най-малък номер на слот получава правото да предава. Съдържанието на същото поле при фрагмент различен от последния е показано на фигура 91.

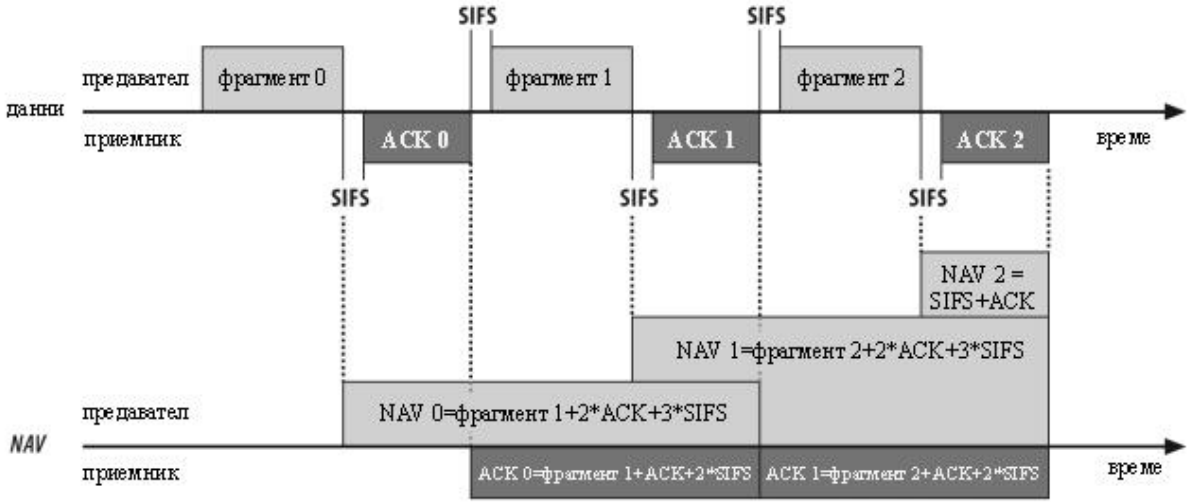


фигура 91 *Полето Duration за фрагмент различен от последния за кадъра*

Когато се налага даден кадър да бъде фрагментиран се сформира т. нар. fragmentation burst. През този период предавателят запазва контрола върху канала за предаване (фигура 92 и фигура 93). NAV гарантира, че другите станции няма да използват канала за предаване през това време.



фигура 92 *RTS/CTS и фрагментация*



фигура 93 *Фрагментация*

6.13.8. Сигурност при Wireless мрежите

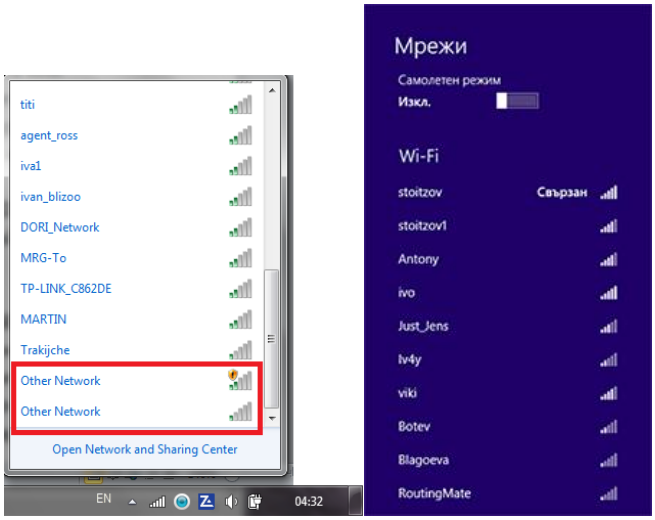
За ограничаване на достъпа до определена безжична мрежа могат да се използват няколко подхода, осигуряващи различна степен на сигурност. Конфигурационните им настройки могат да бъдат установени при администриране на AP устройствата.

- *SSID* (*Service Set Identifier*) - основен механизъм за разграничаване на две различни безжични мрежи. Всички устройства в една *WLAN* трябва да имат един и същ *SSID*. Представлява последователност от символи, които обозначават една *WLAN* мрежа, и дължината му е от 1 до 32 знака (байта);
- *MAC филтриране* - може да се извършва допускане до мрежата в зависимост от *MAC* адреса на клиентското устройство;

- *WEP* (*Wired Equivalent Privacy*) - протокол за сигурност, създаден да предложи защита на данните, съпоставима с тази на една кабелна мрежа. Открит е пробив в сигурността на стандарта, което го прави уязвим за атаки;
- *WPA (Wi-Fi Protected Access*) - стандарт с подобрено криптиране на данните и сигурна автентикация.

6.13.8.1. Скриване на (E)SSID

По подразбиране всеки един безжичен маршрутизатор с активиран безжичен модул е настроен през определен период от време да излъчва пакети (beacons или още маяци), които да информират намиращите се наблизо клиенти, работещи на същия стандарт, за съществуването му. Един такъв пакет-маяк съдържа информация, както за **BSSID** (**MAC** адреса на рутера), така и за **Extended SSID** (или още **SSID** - името на безжичната мрежа).



фигура 94 *Идентификация на Wireless мрежи*

Активирайки опцията за скриване на името на безжичната мрежа (**ESSID**-то) пакет-маяците, които се излъчват от рутера спират да носят информация за нейното име и съответно тази мрежа се превръща в "скрита".

За да може един клиент да установи връзка с безжичната мрежа на маршрутизатора, той трябва да знае няколко нейни основни параметъра - **BSSID**, **ESSID** и честотен канал, на който тя работи. Информация за **BSSID**-то и честотния канал се извлича от пакет-маяците, но при липса на данни за името на мрежата се разчита на всеки един клиент да го попълни

ръчно. Ако **ESSID**-то заявено от клиента е вярно ще се осъществи комуникация с рутера, а при грешно име на мрежата, рутерът ще игнорира съответния клиент.

6.13.8.2. MAC Филтриране

Повечето рутери предоставят възможността да позволят или да забранят достъпа до безжичната им мрежа на определени клиенти. Това се постига чрез използването на филтриране по MAC адрес. За целта се използва списък от MAC адреси. В зависимост от това дали на тези MAC адреси им е позволено или забранено достъпването до безжичната мрежа, рутерът ще обработва или игнорира техните заявки.

6.13.8.3. Статично IP адресиране

Статичното IP адресиране (изключен DHCP сървър) може да се превърне в една допълнителна мярка за сигурност. То не би могло изцяло да попречи на нежеланите клиенти, а само временно да ги затрудни. Това означава, че на него не може да се гледа като метод за защита.

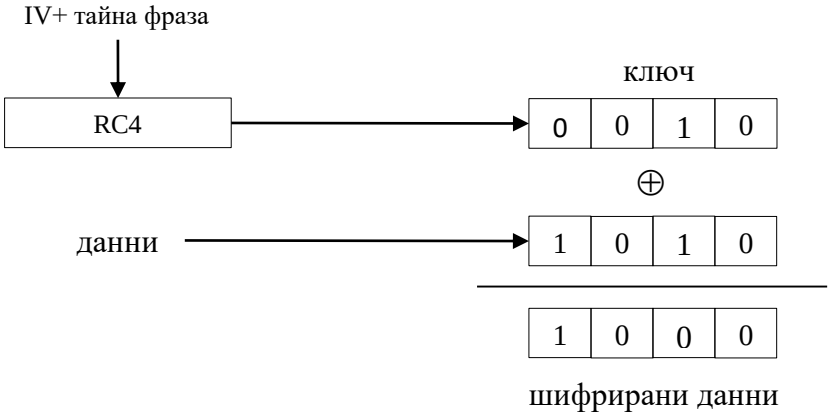
6.13.8.4. WEP Шифриране

Wired Equivalent Privacy (**WEP**) е първия криптографски алгоритъм предназначен за защита и шифриране на данните предавани по стандарта 802.11. Въведен като автентична част от 802.11 и ратифициран през месец септември 1999 година, в наши дни той е считан за несигурен, а след 2004 година е обявен и за непрепоръчителен от IEEE след ратифицирането на един от неговите наследници - **WPA2**. Въпреки това, той все още се поддържа от съвременните маршрутизатори и безжични адаптери с цел съвместимост с по-стари устройства и операционни системи. **WEP** най-често може да бъде срещнат под няколко различни вариации: **WEP-40**, **WEP-104** и **WEP-232**, като числото след "**WEP**-" обозначава бройката битове за използвания шифриращ ключ. Самият ключ винаги се комбинира с уникален инициализационен вектор (**IV**) с фиксирана дължина от 24 бита. Следователно крайната дължина на **WEP-40** е 64 бита, **WEP-104** - 128 бита, а **WEP-232** - 256 бита.

WEP разчита на два алгоритъма - криптографският **Rivest Cipher 4** (**RC4**) и такъв за проверка на грешки **Cyclic Redundancy Check - 32** (**CRC-32**). Двата алгоритъма са линейни по "природа", което в комбинация с

неголемия инициализационен вектор е и причината в **WEP** да бъдат открити множество криптографски слабости.

RC4 е разработен от Рон Ривъст през 1987 година. Причина за вграждането му в **WEP** е простотата и бързината, с която той се изпълнява. **RC4** генерира псевдо-произволен низ от битове (ключ), който се комбинира побитово чрез логическата операция ⊕ с данните, които трябва да бъдат шифрирани (фигура 95).

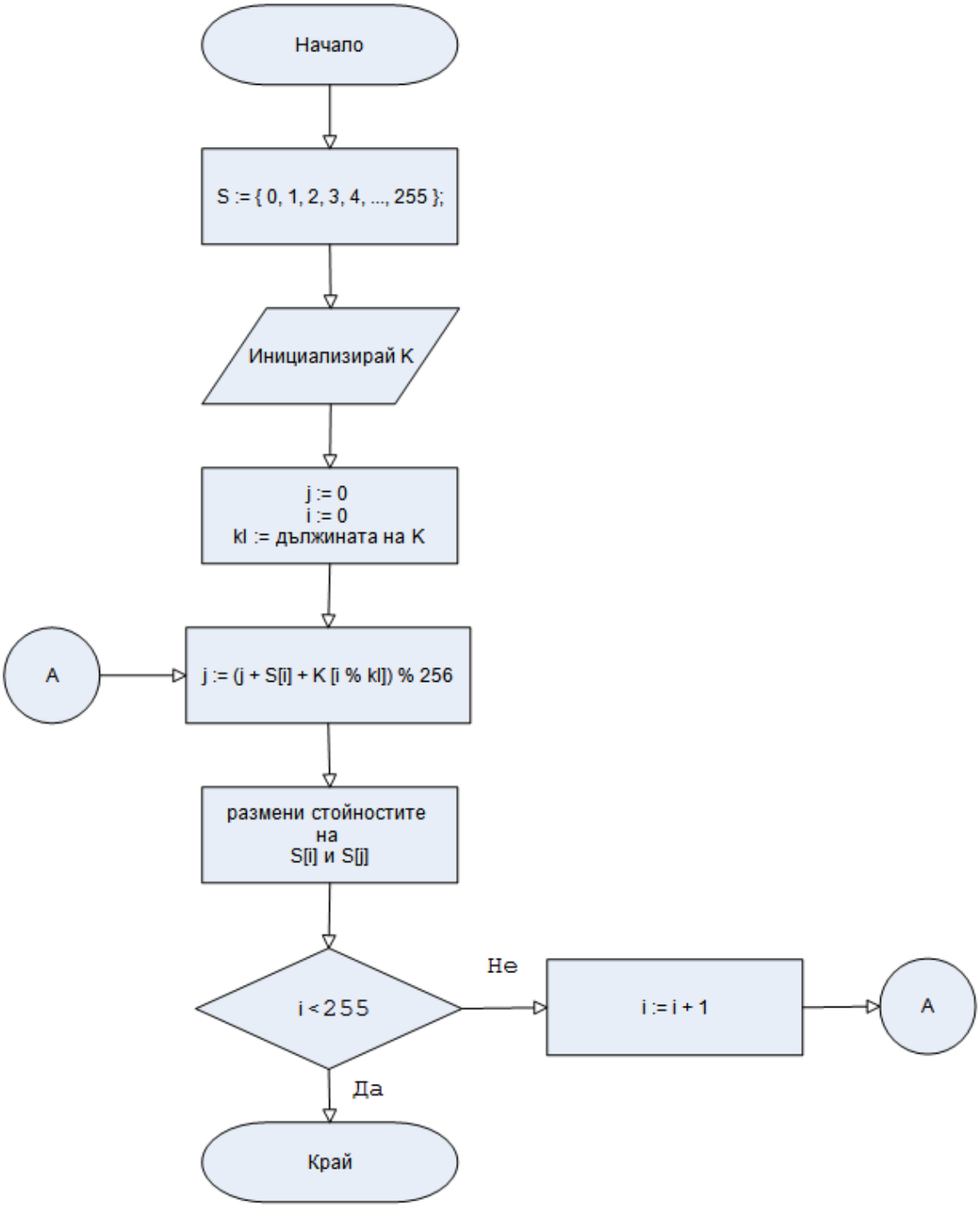


фигура 95 Шифриране

Работата на **RC4** протича през две основни фази:

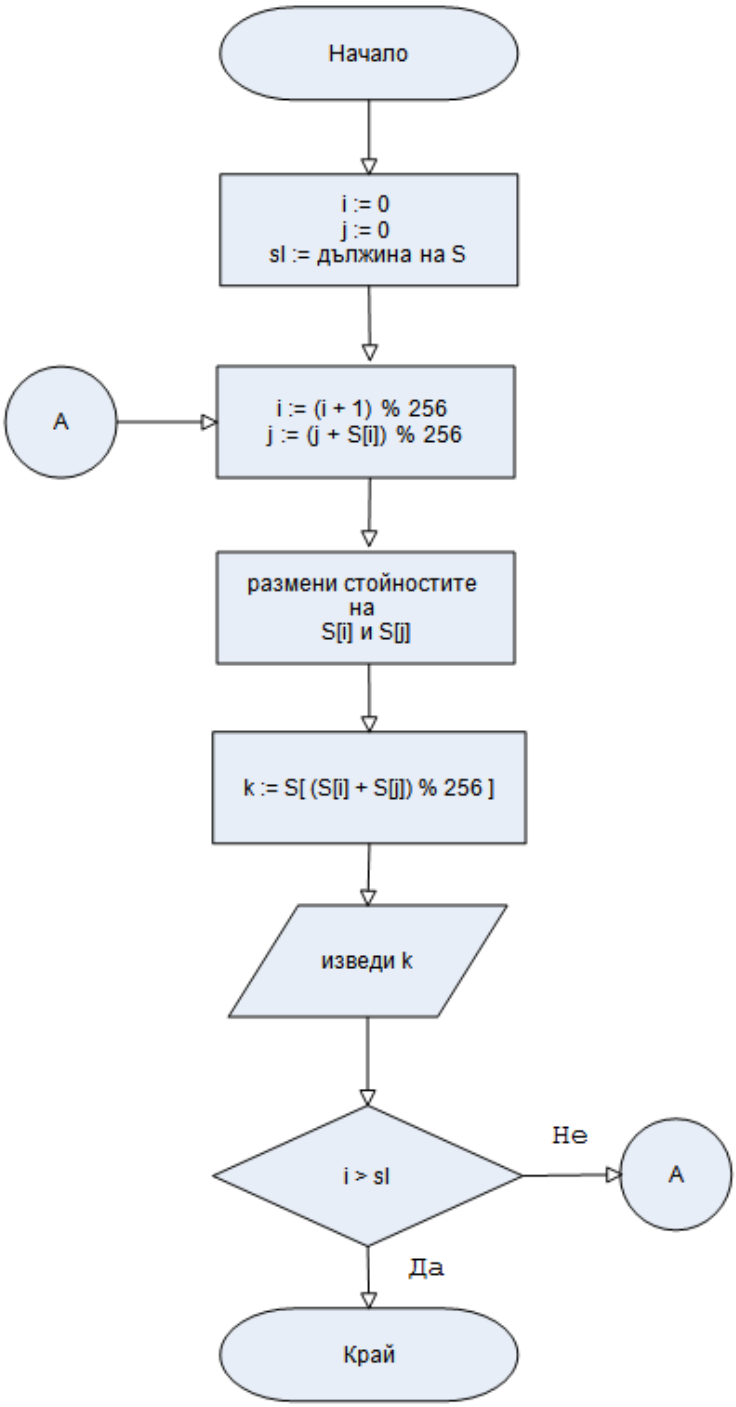
1. Генериране на псевдо-произволен ключ от първоначална "тайна фраза".

Нека вектор **S** с дължина **256** елемента съдържа всички възможни комбинации от 8 битови стойности както следва: **S = { 0, 1, 2, 3, 4, ... , 255}**, а вектор **K** съдържа тайната фраза с максимална дължина **2048** бита.



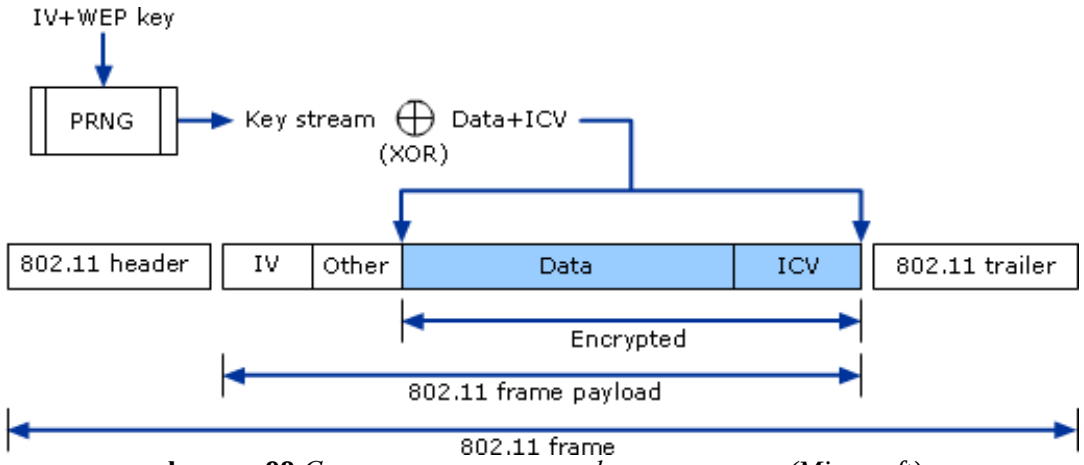
фигура 96 *Генериране на псевдо-произволен ключ*

2. Избор на елемент от ключа за XOR при генериране на крайни шифрирани данни.

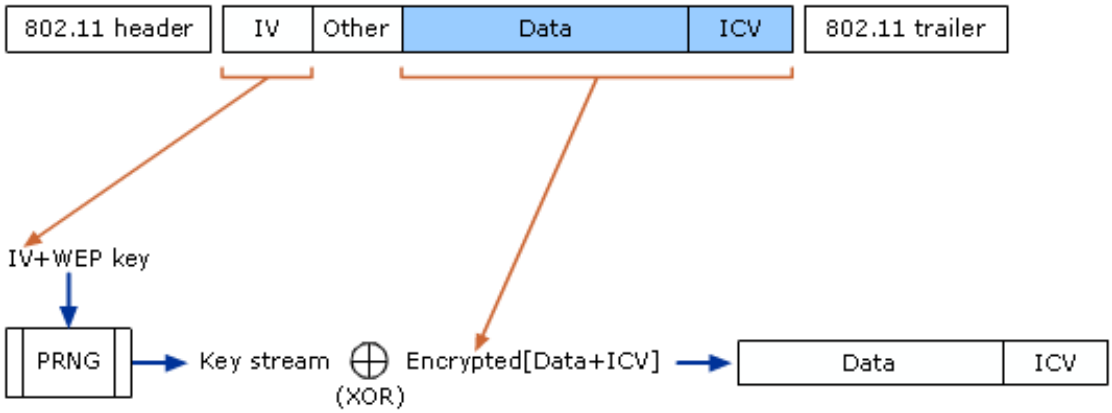


фигура 97 *Графично представяне на избора*

Краят на шифрираните данни съдържа и изчислена CRC-32 контролна сума, наречена (ICV – Integrity Check Value), служеща за проверка на целостта на данните, приети от получателя след тяхното изпращане. След пристигането на пакета до съответния получател, той отново изчислява ICV сумата на пакета и я сравнява с приложената в него. Ако двете не съответстват, това показва че данните от пакета не са били предадени коректно.

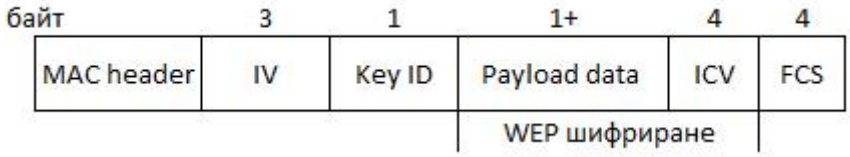


фигура 98 *Схема на цялостен шифриращ процес (Microsoft)*



фигура 99 *Схема на цялостен дешифриращ процес (Microsoft)*

Общ вид на една **MPDU** (MAC protocol data unit) е показан на фигура 100.



фигура 100 *MAC protocol data unit*

Initialization Vector (IV) – уникална 24 битова стойност комбинирана с **Key ID**, за да се сформира ключът използван в **RC4** шифъра.
Key Identifier (ID) – идентификатор на ключа: битове 7 и 6 се използват да съхраняват индекса му; битове 5 – 0 са резервирани за служебни нужди и обикновено са 0.
Payload Data – данните за изпращане.
Integrity Check Value (ICV) – сумата за проверка изчислена върху нешифрираната **Payload Data**.
Frame Check Sequence (FCS) – IEEE 32-bit cyclic redundancy code (**CRC-32**) изчислен от всички горни полета на **MPDU**.

Съществуват два метода на **WEP** автентикация - **Open System** и **Shared Key**. Първият метод позволява на всеки един клиент да се свърже към **WEP** мрежата, без от него да се изисква ключ (парола), а **WEP** ключа се използва единствено за шифриране на безжичния трафик между комуникиращите. При **Shared Key** метода се използва четирифазен модел на оторизация (англ. вариант "four-way handshake"):

1. Клиентът изпраща запитване за автентикация в **WEP** мрежата;
2. Точката за достъп отговаря на клиента с нешифрирано съобщение;
3. Клиентът шифрира полученият отговор със своя **WEP** ключ и го изпраща отново;
4. Точката за достъп дешифрира полученото съобщение със своя **WEP** ключ и ако резултата съвпада с нешифрирания му вариант, тя разрешава достъпа (т.е. за да има съвпадение **WEP** ключовете трябва да съвпадат). При разрешен достъп всяка следваща комуникация между двете страни протича шифрирано.

При този метод ако бъдат "прехванати" съобщенията изпратени по горната процедура те могат да бъдат използвани за определяне на шифриращия ключ по метода на грубата сила.

Въпреки, че WEP притежава множество криптографски слабости, прилагането на метода на грубата сила за откриване на първоначалната тайна фраза е изключително бавна и неефективна техника. Това обаче не е единствения начин, по който той може да бъде разбит. Една от най-популярните и ефективни атаки е **FMS**, кръстена на нейните създатели **Fluhrer, Mantin и Shamir**. **FMS** става достояние на света след като нейната концепция е публикувана през 2001 година. Не след дълго тя е интегрирана и в доста кракерски инструменти.

В същността на **FMS** е идеята за колизиите на инициализационните вектори, които настъпват при интензивен комуникационен трафик между рутер и клиент. Инициализационният вектор, който по принцип би трябвало да е уникален, се изгражда от неголям брой битове, които в един момент се изчерпват и някой вече използван вариант на **IV** може да се повтори отново. Това означава, че в даден момент две напълно различни съобщения ще бъдат шифрирани с един и същ ключ. При настъпването на такова събитие атакуващият може да извлече първоначалния шифриращ ключ.

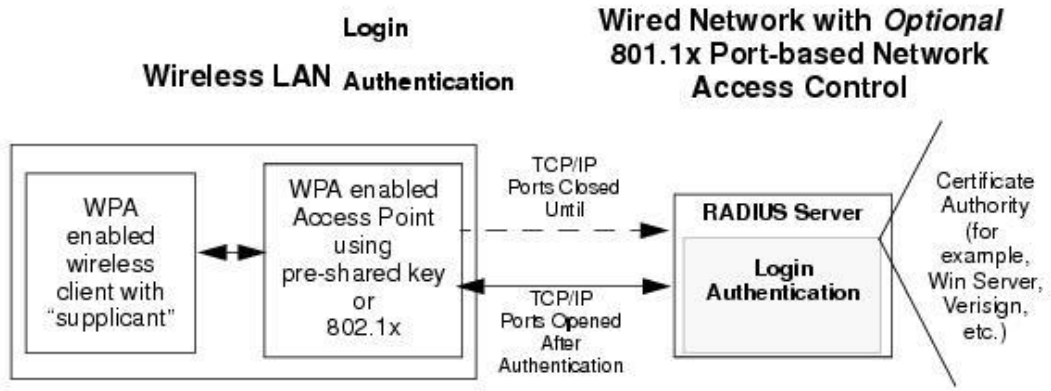
Разликата в ефективността между brute-force и FMS атаките е огромна. При първия вид разбиването може да отнеме дори години, докато FMS средно отнема до около 30 минути, а в някой случаи и под минута в зависимост от интензивността на трафика.

6.13.8.5. WPA шифриране

През 2003 година, след множеството открити пропуски в сигурността на WEP, част от 802.11 стандарта става и алгоритъмът Wi-Fi Protected Access (WPA). Разработен от Wi-Fi Alliance, той е предназначен да замести WEP без да притежава неговите слабости. Използва методи, които се поддържат от по-старите хардуерни устройства след смяна на firmware-а им.

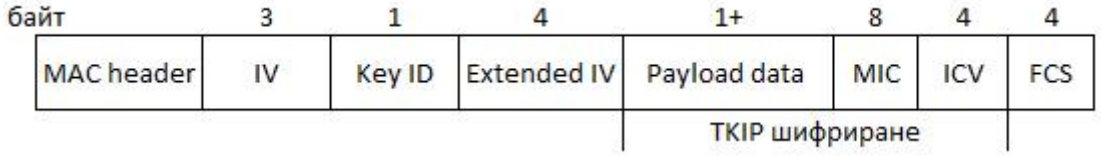
WPA използва шифриране базирано на RC4, наречено Temporal Key Integrity Protocol (TKIP), което разбърква RC4 шифриращия ключ, всеки път когато е получен или е изпратен пакет с данни. Самият TKIP използва 128 битов ключ за шифриране и дешифриране както и 64 битов ключ наречен Message Integrity Code (MIC) за проверка на валидността на изпратените/получените данни. MIC се базира на алгоритъма Michael специално създаден, за да замести използвания в WEP CRC-32. TKIP ключът може да бъде един от двата типа – ключ по двойки (pairwise key) или групов ключ (group key). Първият вид се използва за всички видове пакети изпращани от 802.11 станцията, а вторият за всички мултикаст и броадкаст пакети получени от станцията.

Съществуват две разновидности на WPA – WPA Personal и WPA Enterprise. При WPA Personal известен още като WPA-PSK (Pre-shared Key) единствено точката за достъп определя дали даден клиент е оторизиран да се свързва към мрежата на базата на верността на неговия криптиращ ключ. При WPA Enterprise варианта, рутерът представлява посредник между клиента и RADIUS сървър за автентикация. До RADIUS сървъра може да бъде изпратен Pre-shared Key или да се използва друг вид оторизация – Extensible Authentication Protocol (EAP), която може да бъде EAP-TLS,EAP-TTLS/MSCHAPv2, PEAPv0/EAP-MSCHAPv2, PEAPv1/EAP-GTC, PEAP-TLS, EAP-SIM, EAP-AKA, EAP-FAST.



фигура 101 *WPA Enterprise вариант*

WPA Enterprise се използва по-често в бизнес организациите, а за домашни мрежи се препоръчва WPA Personal.



фигура 102 *MPDU-MAC protocol data unit*

IV – Инициализационен вектор; Байтове 1 и 0 съдържат TKIP Sequence Counter (TSC), използван за защита при повторно изпращане на пакет
Key ID – Битове 7 и 6 съхраняват индекса на ключа. Бит 5 индикира; присъствие на Extended IV. Битове 4 – 0 са резервирани и обикновено са 0
Extended IV – Съдържа байтове 5 – 2 от TSC
Payload Data – Данните за изпращане
Message Integrity Code (MIC) – Сума за проверка, изчислена върху Payload Data чрез Michael алгоритъма
Integrity Check Value (ICV) – Сума за проверка, изчислена върху нешифрираната Payload Data
Frame Check Sequence (FCS) – IEEE 32-bit cyclic redundancy code (CRC-32) изчислен от всички горни полета на MPDU.

WPA и в частност WPA-TKIP остава сравнително сигурен и безопасен за повечето обикновени потребители, чиято цел главно е да ограничат достъпа до безжичната си мрежа посредством парола. Въпреки това, той не се препоръчва за употреба в безжични мрежи, където е необходимо нивото на сигурност да бъде максимално високо. Съществуват по-комплексни атаки позволяващи на атакуващия при специални условия – WPA-TKIP и наличие и поддръжка на QoS от страна на маршрутизатор и клиент, в рамките на 15 минути MIC ключа да бъде дешифриран. Това може да позволи на атакуващия да инжектира свои собствени пакети в мрежата,

които да бъдат използвани за осъществяването на други атаки като например arp poisoning.

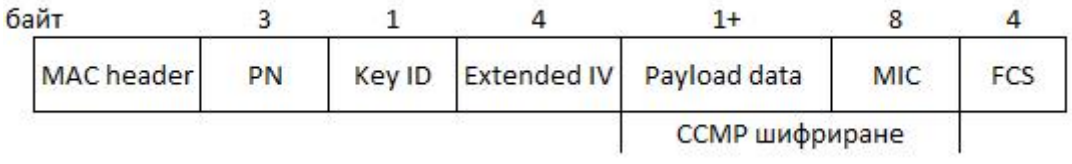
6.13.8.6. WPA2 Шифриране

Една година след излизането на WPA, през 2004 се появява и неговият наследник – WPA2. За разлика от предшественика си WPA2 не е съвместим със старите 802.11 устройства, тъй като начина по който шифрира данните изисква по-специализиран хардуер. Новият метод за шифриране предназначен да замести TKIP се нарича Counter Cipher Mode with Block Chaining Message Authentication Code Protocol или още CCMP (CCM mode Protocol). CCM режимът е бил създаден специално за шифри с дължина на блока от 128 бита. Такъв е и шифриращият алгоритъм Advanced Encryption Standard (AES), който WPA2 прилага. Именно затова WPA2 CCMP е още известен като WPA2 CCMP-AES.

AES-CCMP използва 128 битов ключ за шифриране и дешифриране. Самият ключ може да бъде един от двата типа – ключ по двойки (pairwise key), който се използва за всички пакети изпратени от 802.11 станцията включително уникаст, мултикаст и броадкаст пакети, както и за всички получени от станцията пакети и групов ключ (group key) - за всички мултикаст и броадкаст пакети получени от станцията.

Освен AES-CCMP, WPA2 поддържа и TKIP. По-късно подобно на WPA2 варианта WPA AES-CCMP също добива популярност, но той се поддържа единствено от хардуер, който може да работи и с WPA2.

В WPA2 отново може да се използва Personal (WPA2-PSK) или Enterprise оторизация, както при на WPA.



фигура 103 *MPDU-MAC protocol data unit*

Packet Number (PN) – Съдържа байтове 1 и 0 от AES-CCMP PN стойността, използвани за защита при повторно изпращане
Key Identifier (Key ID) – Битове 7 и 6 съдържат индекса на ключа. Бит 5 показва наличието на Extended IV. Битове 4 – 0 са резервирани и са 0
Extended Initialization Vector (Extended IV) – съдържа байтове 5 – 2 от AES-CCMP PN стойността
Payload Data – Данните за изпращане

Message Integrity Code (MIC) - Сума за проверка, изчислена върху Payload Data на MPDU
Frame Check Sequence (FCS) – IEEE 32-bit cyclic redundancy code (CRC-32) изчислен от всички горни полета на MPDU.

За момента WPA2-CCMP остава най-сигурният шифриращ протокол. Както при WEP и WPA за него е възможно прилагането на bruteforce метод за разкриване на оригиналния шифриращ ключ, но тъй като AES-CCMP е сравнително сложен и изчислението му отнема време, тази техника остава много трудно приложима (в най-лошия случай може да даде краен резултат след няколко милиарда години при използването на изчислителната мощ, с която разполага днешен средностатистически компютър).

6.13.8.7. Wi-Fi Protected Setup

Създаден през 2007 година от Wi-Fi Alliance, Wi-Fi Protected Setup (WPS) е предназначен да позволи на технически незапознатите домашни потребители по лесен и сигурен начин да добавят нови устройства към тяхната безжична мрежа, без да е необходимо да въвеждат пароли (като те се обменят между устройствата по WPS протокола). Съществуват няколко начина на WPS оторизация:

- PIN код – изисква се да бъде въведен (предоставен) от устройството, желаещо достъп до мрежата. PIN кодът може да бъде генериран от клиента или точката за достъп. Подходът зависи от начина на реализация от съответния производител;
- Бутони – потребителят натиска реални и/или виртуални бутони на устройствата, които иска да свърже;
- Near Field Communication (NFR) – технология подобна на RFID, позволяваща безжична обмяна на информация на близки разстояния, обикновено до няколко десетки сантиметра;
- USB – използва се USB памет.

В края на 2011 година са открити слабости в начина, по който се генерира WPS PIN кода и начина на проверката му в процеса на оторизация. В голям процент от случаите това позволява по метода на грубата сила, в рамките на няколко часа, атакуващият да получи достъп до съответната безжична мрежа. Особено уязвими са точки за достъп, които не изискват някаква физическа намеса от страна на потребителя, за да се

стартира WPS процедура. Някой от тях дори не позволяват изключването на WPS или го изключват само привидно, оставяйки устройството напълно уязвимо.

И днес голяма част от рутерите, които се произвеждат пристигат до потребителите с активиран WPS. Тъй като WPS представлява един вече несигурен метод за обмяна на първоначалният шифриращ ключ между рутер и клиент, то неговото получаване не може да бъде затруднено дори и при използване WPA2-CCMP. Единственият начин за решение на проблема е изключването на WPS.

6.13.9. Радио параметри на средата

Децибелът се използва за изразяване на относителната разлика в нивото между два сигнала. Изчислява се по формулата:

(*1) $X_{dB} = 10.\lg\left(\frac{P_2}{P_1}\right)$, където P_1 и P_2 са двете мощности.

Например, ако изменението на мощността $\frac{P_2}{P_1} = 100$ то

$$10.\lg(100) = 20dB.$$

Ако във формула (*1) P_1=1mW (стойност, приета за опорна единица при IEEE 802.11) то резултатът от изчисляването на произволна мощност **P** (изразена във **W**) ще генерира резултат в **dBm**. Резултатните уравнения ще изглеждат по следния начин:

(*2) $Y_{dBm} = 10.\lg\frac{P_W}{1mW} = 10.\lg\frac{P_W}{10^{-3}W} = 10.\lg(1000P_W) = 10.\lg P_W + 30$

(*3) $P_W = \frac{10^{\frac{Y_{dBm}}{10}}}{1000} = 10^{\frac{Y_{dBm}-30}{10}}$, където 1mW=$10^{-3}$W

На базата на цитираните формули таблица 7 визуализира преобразуване на dBm към W.

dBm	W	dBm	W	dBm	W	dBm	W
0	1.0 mW	16	40 mW	32	1.6 W	50	100 W
1	1.3 mW	17	50 mW	33	2.0 W	54	250 W
2	1.6 mW	18	63 mW	34	2.5 W	57	500 W
3	2.0 mW	19	79 mW	35	3.2 W	60	1 000 W
4	2.5 mW	20	100 mW	36	4.0 W	64	2 500 W

5	3.2 mW	21	126 mW	37	5.0 W	67	5 000 W
6	4 mW	22	158 mW	38	6.3 W	70	10 000 W
7	5 mW	23	200 mW	39	8.0 W	74	25 000 W
8	6 mW	24	250 mW	40	10 W	77	50 000 W
9	8 mW	25	316 mW	41	13 W	80	100 000 W
10	10 mW	26	398 mW	42	16 W	84	250 000 W
11	13 mW	27	500 mW	43	20 W	87	500 000 W
12	16 mW	28	630 mW	44	25 W		
13	20 mW	29	800 mW	45	32 W		
14	25 mW	30	1.0 W	46	40 W		
15	32 mW	31	1.3 W	47	50 W		

таблица 7 *Преобразуване на dBm към W*

Пример 1: Стойността на децибелите на предавател с мощност 10mW е 10dBm.

Изчислява се по формула (*2):

$$10.\lg\left(\frac{10\ mW}{1\ mW}\right) = 10.lg10 = 10.1 = 10dBm$$

Пример 2: Стойността на децибелите на усилвател, чието усилване е 10000 пъти, е 40 dBm.

Изчисляването се извършва по формула (*1).

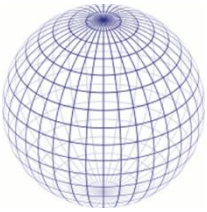
$$10.lg\left(\frac{P_2}{P_1}\right) = 10.lg10000 = 40dBm$$

От подобни изчисления или таблица 7 се получава, че:

- двойно увеличаване на мощността е с означение (3 dBm), 10 пъти увеличение – (10 dBm), 1000 пъти увеличение – (30 dBm);
- наполовина намаляване на мощността е с означение (-3 dBm), 10 пъти намаление – (-10 dBm), 1000 пъти намаление – (-30 dBm);

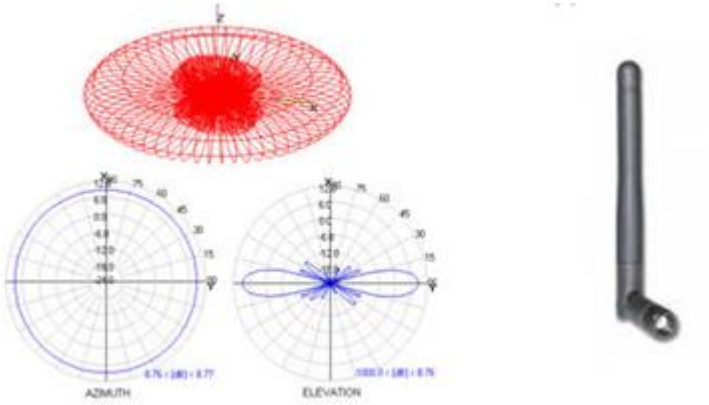
6.13.9.1. Антени

Една от основните характеристики на всяка антена е коефициентът ѝ на усилване, измерван в dBi (i - isotropic), показващ степента на усилване на сигнала спрямо изотропната антена. Тя представлява точка, която излъчва равномерно електромагнитни вълни във всички посоки на пространството.

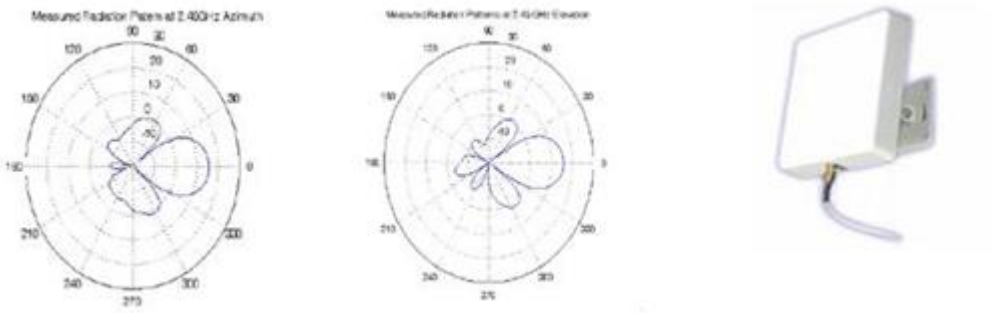


фигура 104 *Сферична форма на излъчваща изотропна антена*

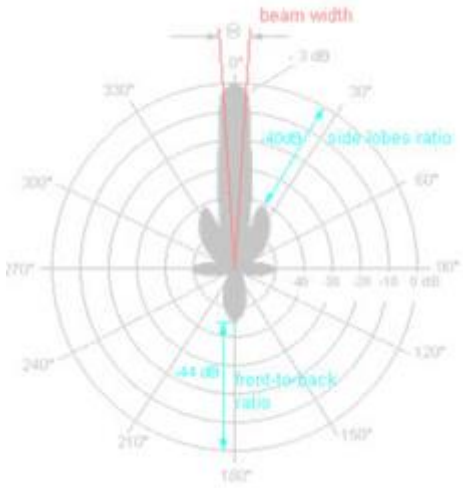
Реалните антени имат диаграми различни от сферичната. Примери за такива диаграми на излъчване са визуализирани на следващите три фигури.



фигура 105 Излъчване на диполна антена

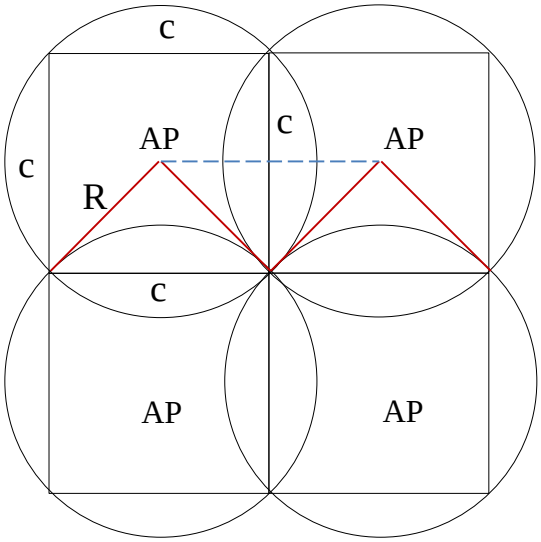


фигура 106 *Излъчване на Patch панел антена*



фигура 107 *Излъчване на насочена антена*

Диаграмите на излъчване на антените показват концентрация на повече електромагнитна енергия в дадена област, в сравнение с изотропната антена, което гарантира усилване на сигнала на тези места. Например, при необходимост от покриване на определена област с диполни антени, трябва да се изчислят точките на разполагането им. Ако се приеме, че радиусът на покритие на такъв тип антена е приблизително 50м то областта се разделя на равни квадратни части, както е показано на.



фигура 108 *Разделяне на област при диполни антени (R≈50m)*

Дължината на зоната на препокриване **c**, може лесно да бъде изчислена по питагоровата теорема: $2.R^2 = c^2$, където R е радиуса на покритие на диполната антена, а **c** е отсечката на препокриване.

Допълнително трябва да се предвидят и фактори като:

- конструкцията на сградата (железобетонните стени, метални врати);
- наличието на помещения, в които не се предполага използване на безжичен достъп (санитарни възли, кухненски и складови помещения);
- наличието на помещения с предполагаема висока концентрация на потребители (чакални, конферентни зали, кафенета, учебни помещения);
- възможни устройства, смущаващи работата на безжичните точки за достъп (предаватели и ретранслатори за мобилни, радио и ТВ комуникации).

6.13.9.2. Физически фактори, указващи влияние върху разпространението на сигнала

Параметрите, имащи значение за разпространението на сигнала са:

1. Мощността на предаване;
2. Загуби по кабела между предавател и антена;
3. Усилване на предавателната антена;
4. Местоположение на двете антени – разстояние, препятствия;
5. Усилване на приемащата антена;
6. Загуби по кабела между антена и приемник;
7. Чувствителност на приемника – минималната мощност на сигнала, зададена в dBm или mW, необходима на приемника за неговото декодиране. Например, чувствителност на приемника от 0 dBm е 1 mW, (-60 dBm) – 0.000001 mW, (-70 dBm) – 0.0000001 mW. Знакът минус означава, че мощността е под опорната стойност от 1mW. Колкото стойността е по-малка (по-отрицателна) толкова чувствителността на приемника е по-добра.

6.13.9.3. Пресмятане на изходната мощност при предаване

Пресмятането се извършва по формулата:

(*4) $EIRP = TX\ Power - Coax\ Cable\ Loss + TX\ Antenna\ Gain,$

където:

EIRP се изчислява в dBm,
TX Power - изходната мощност на предавателя в dBm,

Coax Cable Loss - загуби в свързващия антената кабел в dB,
TX Antenna Gain - усилване на предавателната антена в dBi.

EIRP е стойността, която се контролира от стандартизиращите организации за безжично оборудване в обхват 2.4 GHz или 5 GHz.

6.13.9.4. Пресмятане нивото на приетия сигнал

Пресмятането се извършва по формулата:

(*5) $RX\ Signal = EIRP - FSL + RX\ Antenna\ Gain - Coax\ Cable\ Loss,$
където:

RX Signal - изчислява се в dBm**,**
EIRP – стойността изчислена от (4)**,**
FSL – стойността на затихването в средата в dB,
RX Antenna Gain - усилване на приемната антена в dB,
Coax Cable Loss - загуби в свързващия антената кабел в dB.

Изчисляването на **FSL** е по формулата:

(*6) $FSL = 36{,}6 + 20.lgF + 20lgD,$
където:

F - честотата в MHz,
D - разстоянието между приемника и предавателя в мили.

За гарантиране на нормалната работа на трасето е необходимо да се въведе оперативен запас. Той се изчислява по формулата:

(*7) $SOM = Rx\ signal\ level\ -\ Rx\ sensitivity,$
където:

SOM - оперативен запас в dB,
Rx signal level – ниво на приетия сигнал,
Rx sensitivity – чувствителност на приемника (определя се от техническата спецификация на радио-оборудването)

Нива на SOM:

- SOM ≥ 20dB – напълно достатъчно за гарантирана връзка;
- SOM ≈ 14dB - типична стойност за презапасяване;
- SOM ≤ 10dB – достатъчна при незашумена среда.

Задача:

Да се определи разстоянието, на което може да се изгради линково радио-трасе, ако са налични следните данни за радиомодулите:

1. Мощност на предавателите: 100 mW;
2. Чувствителност на приемниците: -89 dBm;
3. Усилване на антените: 12 dBi;
4. Загуби в свързващите фидери: 1 dB;
5. Работен честотен диапазон: 2.44 GHz.

Допълнителна информация:

1. Мощността на стандартните предаватели (в dBm) е в границите от 18 dBm до 20 dBm (от 63 до 100 mW).
2. Стандартната чувствителност на приемника (в dBm) е между (–75 dBm) и (–100 dBm).

Решение:

1. Изчисляване на изходната мощност в dBm

$$P = 10.lg(100) = 20dBm$$

2. Пресмята се изотропно излъчената мощност чрез (*4)

$$EIPR=TX\ Power-Coax\ Cable\ Loss+TX\ Antenna\ Gain=$$
$$=20-1+12=31dBm$$

3. Пресмята се необходимия сигнал в приемника нужен за правилно декодиране на данните чрез (*7), като се добавя и оперативен презапас от 14dB.

$$RX\_signal = SOM + RX\_sensitivity = 14 + (-89) = -75dBm$$

4. Определя се необходимата стойност за FSL по (*5)

$$FSL = EIRP - RX\ Signal + RX\ Antenna\ Gain - Coax\ Cable\ Loss =$$
$$=31 - (-75) + 12 - 1 = 117dB$$

5. По формула (*6) се определя какво е разстоянието в мили на база на известните параметри:

$$\lg D = \frac{FSL - 36{,}6 - 20.lgF}{20} = \frac{117 - 36{,}6 - 20.lg2440}{20} = 0{,}6326$$
$$D = 10^{0{,}6326} \approx 4{,}29 \text{ мили}$$

След преобразуването в километри се получава:

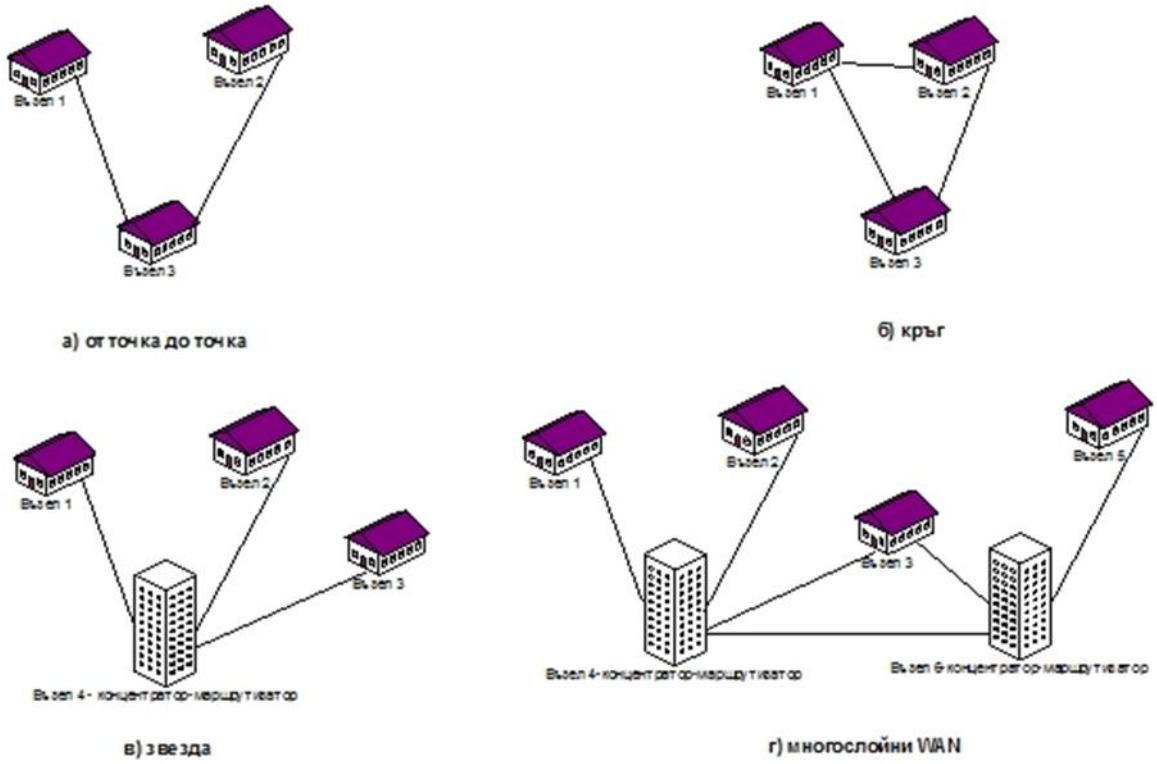
$$D_{km} = D_{miles}.1{,}609344 = 4{,}29.1{,}609344 = 6{,}9\ km$$

От получения резултат може да се заключи, че конкретното оборудване може да постигне ефективно разстояние от 6,9 км.

7. Глобални компютърни мрежи (WAN)

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=24**

Глобалните компютърни мрежи покриват големи географски разстояния. Всяка такава мрежа се състои от множество комутационни възли, свързани помежду им чрез комуникационни линии. Подобно на LAN и тук съществуват топологии. Най-често цитираните топологии са: от точка до точка, кръг, звезда и многослойни мрежи (фигура 109).



фигура 109 *WAN топологии*

При точка а) и б) е показана свързаност между три възела, докато при в) и г) участват допълнителни устройства, наречени концентратори-маршрутизатори, организирани каскадно (г).

Традиционно глобалните мрежи използват или комутация на канали, или комутация на пакети. В учебното съдържание са включени следните технологии:

- комутиране на пакети – X.25, Frame Relay, ATM, B-ISDN [96][95][100];
- комутиране на канали – PSTN, ISDN, xDSL, T-carrier [99][100][94].

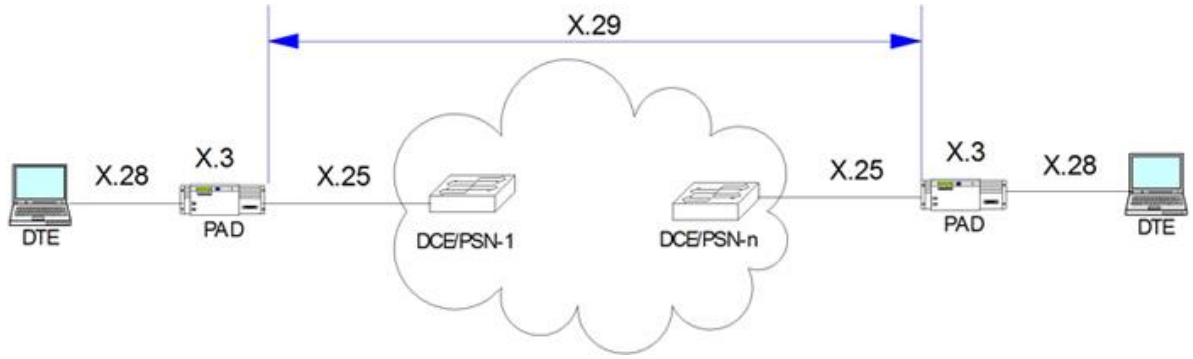
7.1.Стандарт X.25

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=25**

Най-известният и използван стандарт при WAN с комутация на пакети. Предаването на данни е със скорост 2Mbs. Специфицира интерфейса между крайния възел (DTE) и мрежовия комунитационен възел (DTE) от глобалната подмрежа, непосредствено свързан с него.

7.1.1. X.25 устройства

- DTE (Data terminal Equipment) – клиентско устройство, специален пакетен терминал. При обикновен компютър или терминал се поставя PAD (Packet Assembler/Disassembled) устройство между DTE и DCE, изпълняващо три основни функции – буфериране, слепване и разделяне на пакети (фигура 110)
- PSE (Packet-Switching Exchange) или PSN (Packet-Switching Node) – междинен мрежов възел;
- DCE (Data Circuit-terminating Equipment) – PSN, непосредствено свързан с DTE;
- PAD (Packet Assembler/Disassembled) - устройство между DTE и DCE. Работи на нивото на мрежовия слой. Изпълнява и концентриращи функции. За неговото функциониране се използват препоръките “Triple X”:
- X.28 – определя интерфейса между DTE и PAD;
- X.3 – определя услугите на PAD;
- X.29 – описва взаимодействието между две PAD устройства.

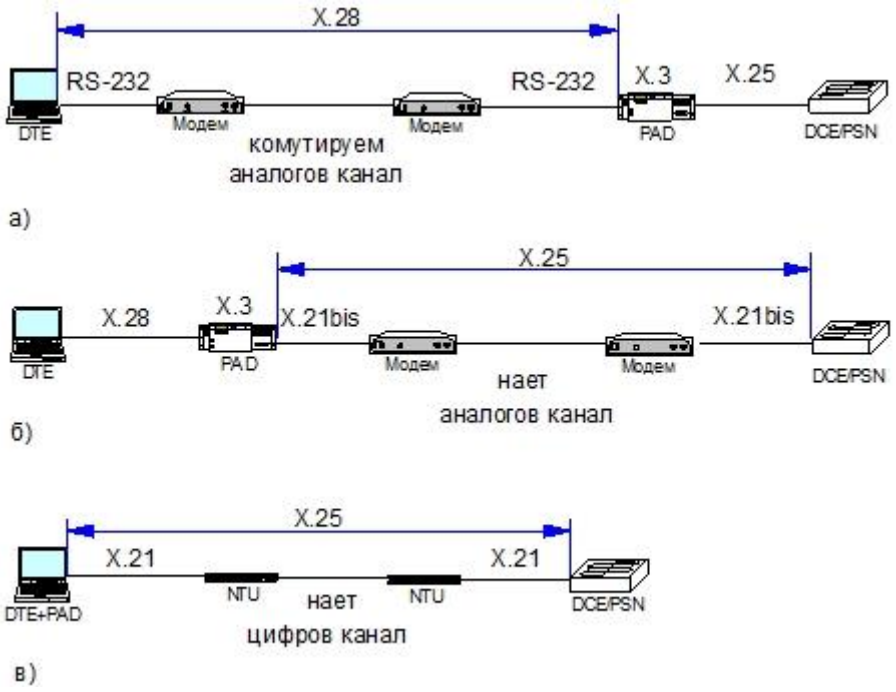


фигура 110 *Triple X*

Разположение на PAD спрямо DTE (фигура 111):

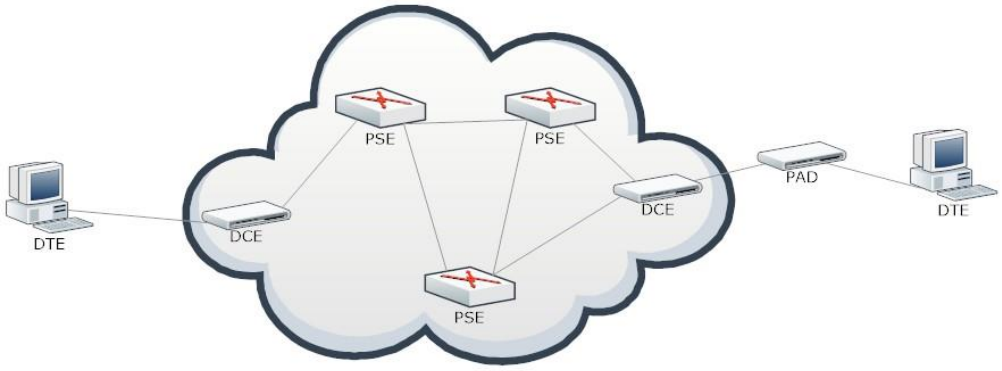
- обществен PAD – DTE се свързва чрез модем (фигура 111а);
- външен PAD – непосредствено до DTE (фигура 111б);

- вътрешен PAD – разположен в DTE (фигура 111в);



фигура 111 *Видове PAD и разположението им спрямо DTE*

Стандартът X.25 е бавен, защото се извършва проверка на всички пакети във всеки PSE възел, което утежнява тяхната работа. Дължи се на тогавашните слаби компютри и ненадеждни аналогови телефонни линии.



фигура 112 *X.25 мрежа*

7.1.2. Протоколен стек

Отнасянето на стандарта X.25 му към OSI модела се ограничава до трето ниво (таблица 8).

OSI модел	Слой	Протоколен стек X.25
Мрежов	3	X.25
Канален	2	LAPB

Физически	1	X.21,X.21bis

таблица 8 *Разпределение на протоколите спрямо OSI модела*

На физическо ниво основните протоколи (интерфейси), които функционират са:

- X.21 - използва цифров интерфейс, цифрова адресация и правила за достъп до наети цифрови линии;
- X.21bis - определя електрически и механични процедури за използване на физическата среда за работа със смесени канали - аналогови и цифрови. Поддържа синхонно пълнодуплексно предаване по съединение тип точка - точка по четирипроводна линия при максимално разстояние между DTE и DCE – 15м.

На ниво канален слой стандартът използва протокол LAPB (Link Access Procedure, Balanced). Основните му функции са свързани с инициализиране на връзката по виртуалната верига, следене за грешки в кадрите и спазване на последователността им, разпадане на връзката. Този протокол проверява всеки приет кадър за грешки като използва цикличен CRC код. Процедурите за контрол се дублират и в мрежовия слой. Това намалява значително вероятността за грешки в каналите. Реализира метода на “плъзгащия се прозорец” (8 кадъра в стандартен и 128 в разширен режим). LAPB използва 3 вида кадри:

- Информационни (I-кадри) – пренасят информацията на горния слой;
- Супервайзорни (S-кадри) – управляват основните функции на LAPB протокола като: потвърждение за приети I-кадри; заявка за повторно предаване на I-кадри; временно задържане на предаване; доклад за състоянието на канала. Не включват полето за данни;
- Неномерирани (U-кадри) – изпълняват допълнителни управляващи функции (преминаване между разширен и стандартен режим, генериране на съобщения за протоколни грешки).

Форматът на кадъра е представен на фигура 113. Полето за контрол е свързано с определяне вида на кадъра.

Flag 01111110 8 бита	Address 8 бита	Control 8 бита	Data променлива дължина	Checksum 16 бита	Flag 01111110 8 бита

фигура 113 *Формат на LAPB кадър*

На мрежово ниво функционира протоколът X.25 (PLP – Packet Layer Protocol), използващ режим на виртуално съединение. Двата вида съединения, поддържани от протокола са:

- PVC (Permanent Virtual Circuits) – постоянни;
- SVC (Switched Virtual Circuits) – комутируеми – протичащи на три фази: установяване на логическо съединение между комуникиращите се обекти, пълнодуплексно предаване на данни и разпадане на логическото съединение.

Основните характеристики на двата вида съединения са представени в таблица 9.

PVC връзка	SVC връзка
Еднократно установяване на връзката	Установяване на връзка при необходимост
Съществува дори и да не се използва	Разпадане на връзката при липса на предаване
Постоянни характеристики	Променливи характеристики
Трудни за управление	Лесни за управление
Строга мрежова архитектура	Гъвкава мрежова архитектура
Един и същи път за PDU	Пътят на PDU може да е различен
Плаща се определена месечна такса	Плаща се само използвания трафик

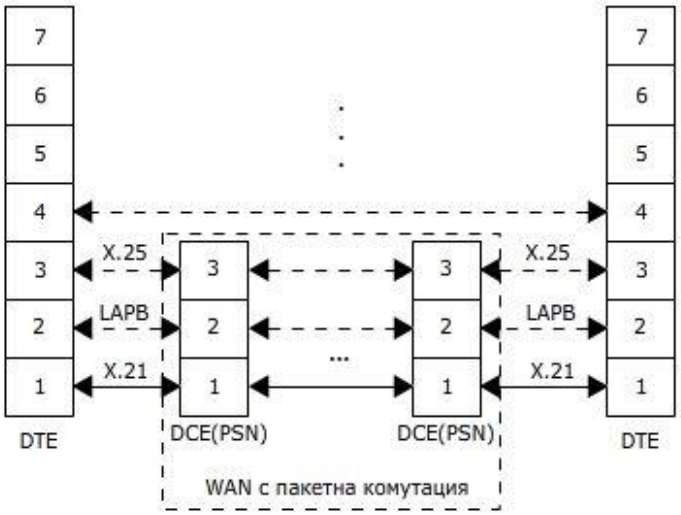
таблица 9 *Сравняване на PVC и SVC*

Форматът на пакета е представен на фигура 114. Някои от полетата могат да бъдат разтълкувани побитово, тъй като всеки бит или съвкупност от битове имат различно значение.

GFI 4 бита	LCI 12 бита	PTI 8 бита	Data променлива дължина

фигура 114. *PLP пакет*

GFI (General Format Identifier) – съдържа побитова информация за управление на потока данни.
LCI (Logical Channel Identifier) – формира номера на логическия канал.
PTI (Packet Type Identifier) – идентификатор на типа на пакета.



фигура 115. *Стандарт X.25*

7.2.Стандарт Frame Relay

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=26**

Frame Relay е наследник на X.25 за достъп до глобална мрежа с комутация на пакети. При Frame Relay имаме опростяване на протоколите, като по-голямата част от обработката се предоставя на крайните възли. Базира се на надеждните цифрови линии, оптична среда и бързи компютри.

Frame Relay може да се разглежда като виртуална наета линия. Абонатите наемат постоянно (PVC) или комутируемо виртуално съединение (SVC) между две свои точки. Всяко PVC съединение представлява фиксиран маршрут между два крайни възела (абоната) на мрежата и има 10-битов номер, означен като DLCI (Data-link connection identifier). Предават се Frame Relay кадри. Скорост – 34Mbs (Европа), T-1 1.544Mbs и T-3 45Mbs (САЩ).

Една от основните разлики между обикновената и виртуалната наета линия е в заплащането. При обикновената наета линия се заплаща договорената максимална скорост на предаване, дори да не се използва. Докато, при виртуалната наета линия се заплаща договорената средна скорост на предаване, като на моменти абонатът може да вдигне и по-голяма от средната.

При виртуалните съединения доставчикът и абонатът се договарят за три параметъра (T_c, B_c, B_e).

- T_c-времетраене на интервалите, на които се разделя времето;
- B_c(Committed Burst Size) – гарантирани байтове за клиента от доставчика за времето T_c;
- CIR=B_c/T_c–договорена или средна скорост (Committed Information Rate);
- B_e(Excess Burst Size)– максимално позволените байтове над B_c, като само B_e се приемат за интервала T_c, но се маркират като нископриоритетни;
- $(B_c+B_e)/T_c$ – максималната скорост, с която абонатът може да предава.

На фигура 116 е представено поведението на честотната лента при нормално и увеличено натоварване.

а) б)

фигура 116. *Договорена информационна скорост (CIR)*

В повечето литературни източници се прави сравнение между виртуалните линии, използвани от стандарта и наетите телефонни линии. Причината за това е конкурентостта между тях. При Frame Relay предимствата са: по-евтина алтернатива, възможност за ползване на Интернет портал (услуга на мрежата), пренасочване на трафика при пропадане на връзката, използва CIR характеристика, позволява увеличаването на честотната лента в натоварени моменти (използва статистическо времеделение), тарифирането е на база трафик.

Междинните мрежови възли са с опростени функции, свързани главно с определяне на границите на кадрите и откриване на грешки в тях. Сгрешените кадри се “бракуват”. Възстановяването им зависи от трафика:

- *при изохронен(видео и глас)* – няма смисъл сгрешените кадри да се предават, защото са чувствителни към закъснение. Резултатът е свързан с пропадане на гласа при гласово предаване;

- *при анизохронен* – не е чувствителен към закъснение, а към грешки. Сгрешените кадри се изискват отново.

Frame Relay не потвърждава правилно приетите кадри за междинните мрежови възли (повишава се бързодействието). Комутаторите “бракуват” кадрите в два случая:

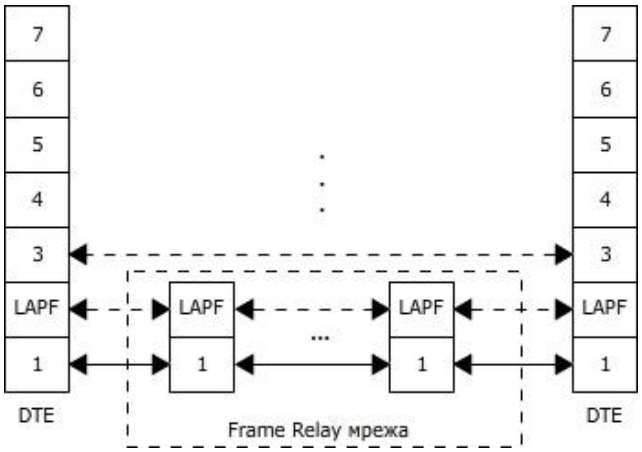
- ако ги приемат с грешка;
- ако не могат да ги съхранят поради препълване на буферите си – комутаторът известява за това крайните възли на всички активни PVC към него.

Стандартът е проектиран за долните две нива на OSI модела (фигура 117).

OSI модел	Номер на слой	Frame Relay
Канален	2	LAPF
Физически	1	I.430,I.431

фигура 117. *Frame Relay и OSI*

На ниво канален слой Frame Relay използва протокола LAPF (Link Access Procedure for Frame Relay). Нивата на функциониране на мрежата са показани на фигура 118.



фигура 118. *Структура на Frame Relay*

Форматът на кадъра е представен на фигура 119.

фигура 119. *Формат на LAPF кадър*

DLCI (Data-link connection identifier) - число, което идентифицира номера на виртуалната верига(VC), между крайното и междинното устройство. Полето позволява до 1024 PVC по една физическа линия. По-голямата част (около 1000) са за абонатите, а останалите се използват за управление на връзката.
C/R – маркира важността на адреса.
EA (Extended Address) – 1 бит. Ако стойността му е 1 то DLCI е само 10 бита. В противен случай се включва и следващото DLCI (разширява се адресното пространство).
FECN (Forward Explicit Congestion Notification) – информира напред за претоварване (предназначен за протоколите за управление на потока данни на получателите).
BECN (Backward Explicit Congestion Notification) – информира назад за претоварване (предназначен за протоколите за управление на потока данни на подателите).
DE (Discard Eligibility) – маркира фрейма като допустим за изхвърляне при липса на ресурси в мрежата. Установява се от DTE устройството.

Стандартът се използва:

- за свързване на локални мрежи;
- като глобална мрежа за анизохронни данни;
- като средство за достъп до ATM мрежи.

Допълнително към стандарта е разработен и т. нар. Local Management Interface (LMI). Като функционалности са добавени:

- поддържане на множество активни виртуални вериги в едни и същи физически крайни точки;
- глобална адресация - DLCI приема стойности с глобално значение, което гарантира уникална идентификация на DTE устройството;
- съобщения за състоянието на виртуалните вериги, осигуряващи комуникация и синхронизация между DTE и DCE устройства. Предотвратяват изпращането на данни по несъществуващи PVC;

- използване на мултикаст групи, целящи ефективно използване на честотната лента при обмен на съобщения между междинните устройства.

7.3.Стандарт ATM (Asynchronous Transfer Mode)

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=27**

Асинхронният трансферен метод (*Asynchronous Transfer Mode- ATM*) е създаден в резултат на дейността по разработката на широколентови цифрови мрежи с интегрирани услуги (*Broadband Integrated Services Digital Network- BISDN*), предназначени за предаване на глас, видеокартина и данни с висока скорост [72][16][6]. Стандартът поддържа скорости 25 Mbps, 155 Mbps и 622 Mbps. Реална е възможността за скорост до 10Gbps, но за сметка на скъпо оборудване. Подобно на X.25 и Frame Relay, ATM използва виртуални вериги (PVC или SVC). Използваната топология е тип "звезда". ATM комутаторът представлява централен възел в мрежата, към който директно се включват всички останали устройства. Това улеснява въвеждането на промени в конфигурация на мрежата и откриване на грешки. ATM използва комутация на пакети и мултиплексиране на няколко логически съединения по един физически интерфейс. Възползва се от предимствата на новопоявилите се цифрови линии. Скоростите са няколко пъти по-високи от Frame Relay.

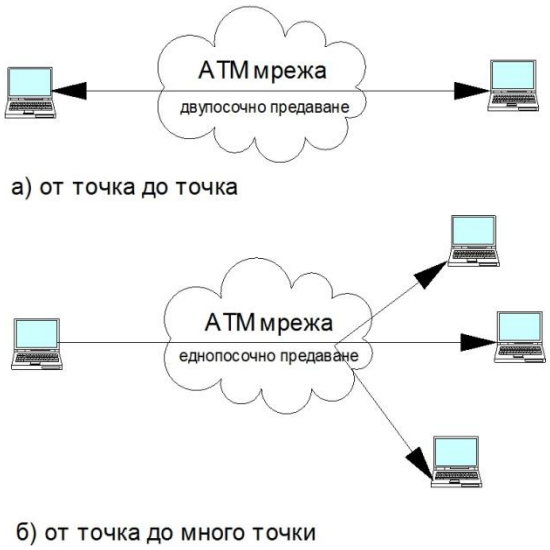
Някои от основните предимства на ATM са следните:

- Ефективно използване на честотната лента- чрез разрешение на достъпа на клиента до мрежата винаги, когато ресурсите са свободни, се осигурява по-добро използване на честотната лента при предаване на данни с голям обем.
- Гъвкавост- осигурява се възможност за използване на широка гама от скорости на предаване и приложения. Разработените интерфейсни стандарти поддържат скорости на предаване от 1,5 Mb/s до 1,2 G/ps.
- Прозрачността на приложенията – методът осигурява трафик, съобразен със скоростта и степента на обем, изисквани от приложенията (данни, глас и др.), а не със скоростта, подходяща за мрежата.

- Мрежови предимства – бърз и прост маршрутизиращ процес, който се базира на виртуалния канален идентификатор (*VCI*), намиращ се в заглавна част на клетката. В мрежата над ниво "*клетка*" други комутационни и маршрутизиращи процеси не се изпълняват, което опростява и увеличава скоростта на обработка на съобщенията. Тази опростена и бърза обработка на съобщения, позволява да се създават високоскоростни самомаршрутизиращи се комутатори, които могат да се разширяват по размер и скорост в съответствие с бъдещи изисквания.

Поддържаните типове връзки са:

- от точка до точка, включващи двупосочно предаване (фигура 120а);
- от точка до много точки, поддържащи еднопосочно предаване (фигура 120б).

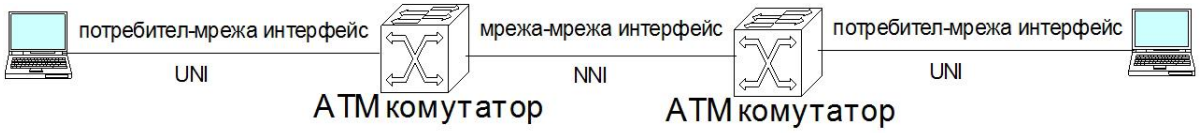


фигура 120. *Типове връзки в ATM*

Двата най-известни интерфейса за стандарта са:

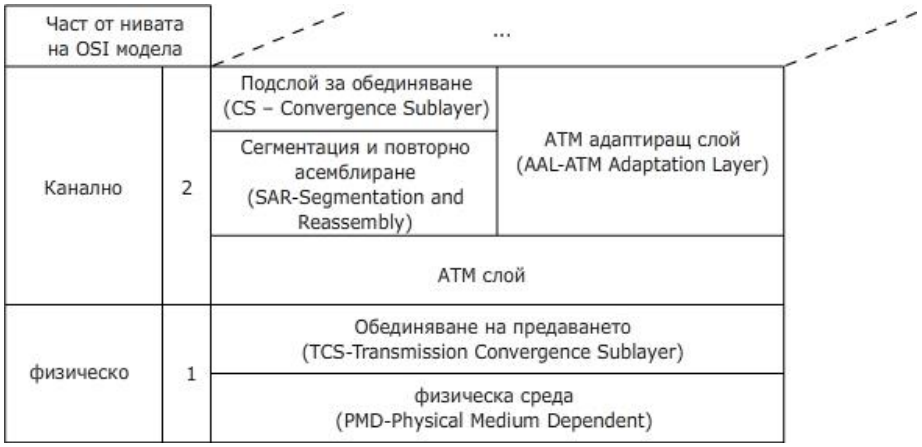
- Потребител-мрежа (User to Network Interface-UNI);
- Мрежа-мрежа (Network to Network Interface-NNI).

Интерфейсите и общия вид на този тип мрежа е илюстриран на фигура 121.



фигура 121. *ATM интерфейси*

Понеже ATM е част от B-ISDN модела, той има пространствена структура. Разпределението на слоевете му във вертикалната равнина и отнасянето към OSI модела е показано на фигура 122 [7].



фигура 122. *Слоеве в ATM модела*

Физическото ниво на стандарта е съставено от два подслоя:

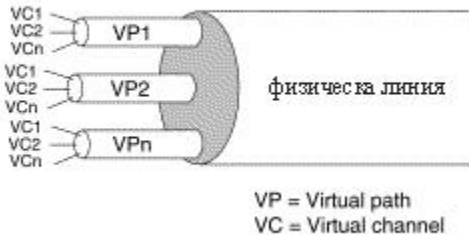
- PMD (Physical Medium Dependent) – обхваща предаването по физическата среда. В този слой се дефинират техническите параметри на използваните куплунги и среди на предаване и се изпълняват функции по кодиране/декодиране, синхронизация и съгласуване;
- TCS (Transmission Convergence Sublayer) – преобразува потока от ATM клетки в поток от битове за предаване по физическата среда и обратно. При наличие на данни за предаване в мрежата се изпращат клетки. Ако липсват данни за предаване, в мрежата трябва да се продължат да се изпращат клетки, за да се запази нейната работоспособност. В този случай подслоят TCS вмъква празни клетки при предаване и отделя празните клетки, когато не достигнат своето местоназначение. Поради различните начини за свързване към оптични или други физически реди за предаване подслоят TCS зависи от съответната среда.

ATM слоят се намира на канално ниво (фигура 122) и е независим от физическата среда. Основни функции на този слой са:

1. Добавяне на заглавна част към SAR данните (фигура 126) в посока към физическия слой и обратната операция при движение на данните към ALL слоя;

2. Механизъм за управление на потока данни при връзката „абонат-мрежа“ чрез полето GFC (Generic Flow Control), принадлежащо на заглавната част на клетката (фигура 127б);

3. Комутация на клетки в междинните възли. Процесът включва пренасочване на данните от дадено входящо виртуално съединение към определено изходящо такова. Този тип съединения се характеризират с номер на физическата линия (номер на порта на комутатора) и номер на логическата линия, за която се специфицират идентификатор на виртуален път VPI (Virtual Path Identifier) и идентификатор на виртуален канал VCI (Virtual Channel Identifier) (фигура 123);



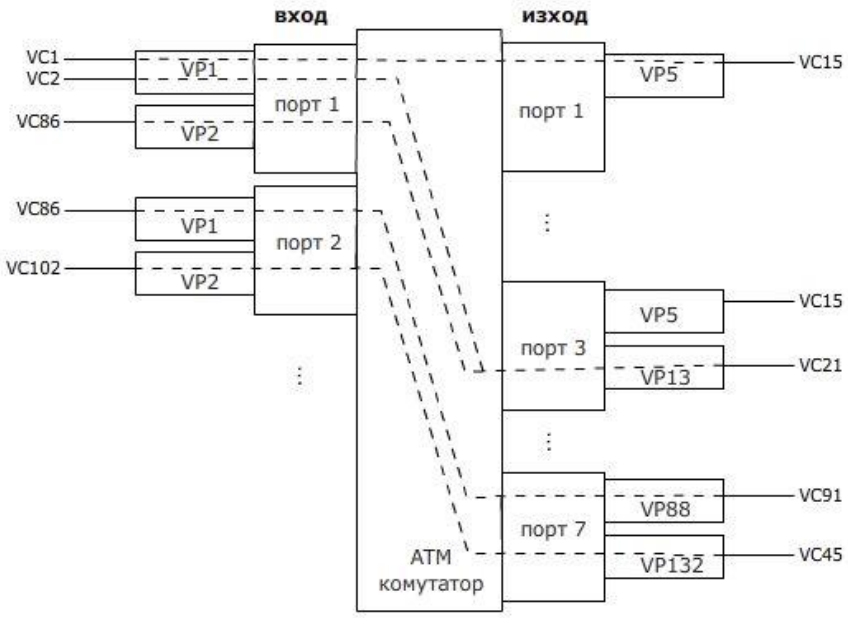
фигура 123 *VPI и VCI*

Виртуалният път е логическа група от виртуални канали, които се маршрутизират заедно през ATM мрежата. Стандартът позволява няколко виртуални пътища да бъдат предавани по една физическа линия. Ограничението на бройката зависи от типа на интерфейса. За интерфейса „потребител-мрежа“ стойността е 256(2^8), а за интерфейса „мрежа-мрежа“ стойността е 4096(2^{12}). Виртуалните канали включени към конкретен виртуален път мога да бъдат до 65536(2^{16}). За тях отговарят полетата VPI и VCI от формата на ATM клетката представена на фигура 127. ATM комутаторите поддържат таблици за актуалните съединения, на базата, на които извършват комутационни действия. Пример за такава таблица е таблица 10, която е илюстрирана чрез фигура 124.

Входен порт	VPI/VCI	Изходен порт	VPI/VCI
1	1/1	1	5/15
1	1/2	3	13/21
1	2/86	3	13/21

2	1/86	7	88/91
2	2/102	7	132/45

таблица 10 *Примерна комутационна таблица на АТМ комутатор*



фигура 124 *Комутация при АТМ комутатор*

4. Мултиплексиране на АТМ клетки от различни виртуални съединения при предаване към физическия слой;

5. Преобразуване на идентификатора на виртуално съединение при комутацията на клетките в междинните възли;

6. Осигуряване на възможност за самите потребители да назначават приоритет на своите клетки чрез CLP (Cell Loss Priority) бит от заглавната част на клетката (таксува се на по-ниска цена).

ALL (ATM Adaptation Layer) слоят адаптира предаваните данните към изискванията на протоколите от по-горния слой. Дели се на два подслоя:

- Горен CS (Convergence Sublayer) подслой – конвертира потокът от данни в CS блокове с подходящ формат съответстващ на определен вид трафик (общо пет вида). За обслужването на отделните видове трафик са планирани пет различни AAL подхода, номерирани с цифрите от 1 до 5. Всеки от тях е проектиран да поддържа един от четирите класа телеуслуги на АТМ. Тази класове са означени с имената A, B, C, D.

- Клас А – използват AAL 1. Услугите от този клас следят две характеристики: постоянната скорост на битовете (Constant Bit Rate - CBR)

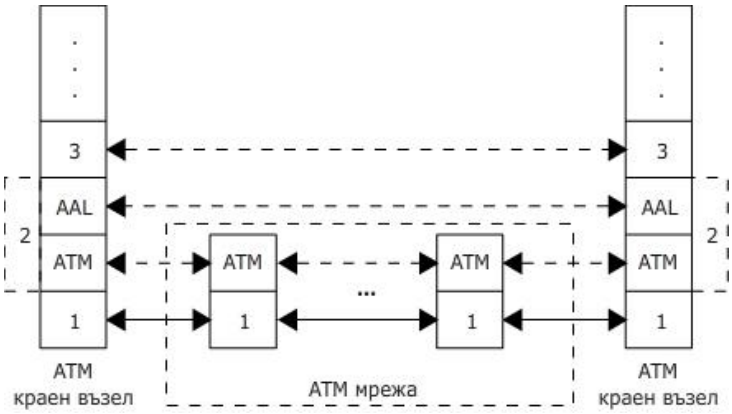
и синхронния трафик във връзката. Необходими са за поддръжка на равномерни комуникации, като глас и звук. Тази постоянна скорост се осигурява от приложението;

- Клас B – подобен на клас A, само че не изисква постоянна скорост на битовете. Използва се променлива скорост (Variable Bit Rate - VBR). Приложенията използват висока скорост на предаване, предаванията са синхронизирани, но те стават през различни интервали от време. Използва се протокол AAL 2, които обаче остава като недовършен проект;
- Клас C – използва AAL 3. Предназначен е за трафик на пакети с променлив размер, предавани по предварително установено логическо съединение;
- Клас D – характеризира се с асинхронен трансфер на данни, независим от връзката. Обслужва се от протокола AAL ¾ (обединение на AAL 3 и 4). Разликата между C и D, е че единият е зависим от връзка, а другия не. За избягване на сложността на протокола AAL ¾ е разработен протоколът AAL 5, който да поддържа основно трафика от клас C.

На практика функционират протоколите с номера 1, ¾ и 5, където 3 и 4 са обединени в означението ¾.

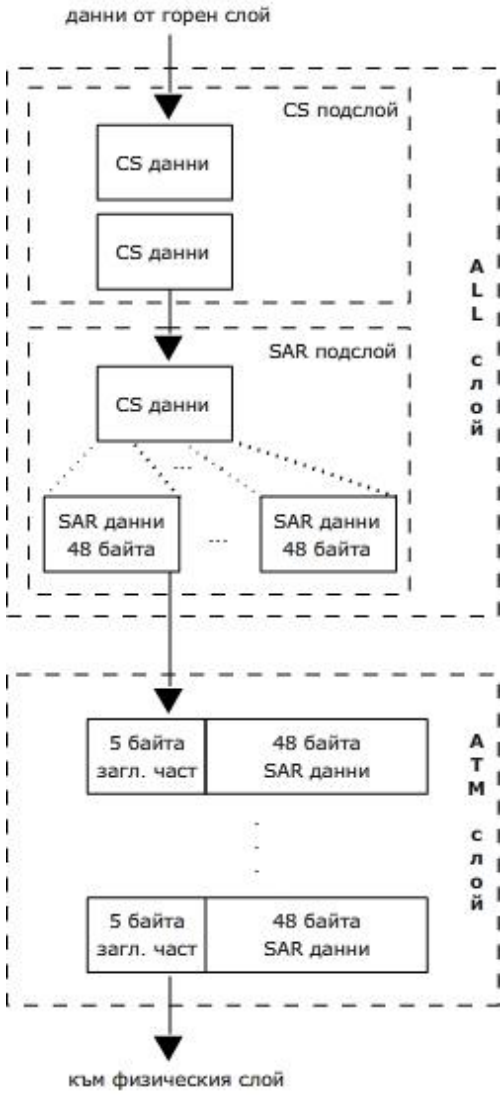
- Долен SAR (Segmentation and Reassembly) подслой – основна функция е сегментиране на данните от CS подслоя.

Функционалните нива на мрежата са визуализирани чрез фигура 125 [7].



фигура 125. *Структура на ATM*

Информационният поток между поднивата на канално ниво (ниво 2) е изобразен на фигура 126.

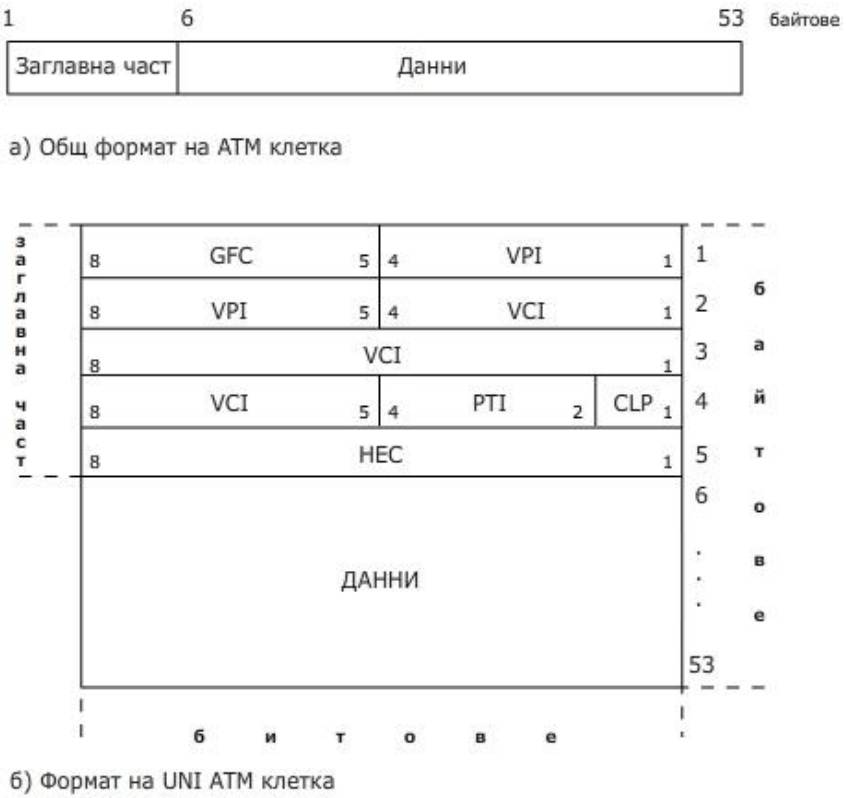


фигура 126. *Информационен поток на канално ниво в ATM*

ATM използва блокове с фиксирана дължина (53 байта), където първите 5 байта са за заглавна част и 48 байта за данни. Комутират се с висока скорост. ATM клетките имат две разновидности:

- UNI – използват се при интерфейса “потребител-мрежа”;
- NNI – между междинните мрежови възли.

Общата структура на клетката [7] може да бъде представена чрез фигура 127а. Форматът на UNI клетка, която лесно се превръща в NNI чрез приобщаване на GFC към VPI полето, е изобразен на фигура 127б.



фигура 127. *ATM клетка*

GFC (Generic Flow Control)- с дължина 4 бита и е предназначено за общо управление на обмена. То може да се използва за реализиране на локални функции, като например идентифициране на няколко станции, които заедно работят с един и същ ATM интерфейс. В типичните случаи полето GFC не се използва и в него е записана стойност по подразбиране.
VPI (Virtual Path Identifier) – с дължина 8 бита и в него е записан идентификатор на виртуалния път. То се използва съвместно с полето VCI за определяне н следващото местоназначение по пътя на клетка, която преминава през серия от ATM комутатора до нейното крайно местоназначение.
VCI (Virtual Channel Identifier) - с размер 16 бита, които представляват идентификатор на виртуалния канал. Използва се заедно с полето VPI от ATM комутаторите за определяне на следващото местоназначение на клетките, постъпили на входните интерфейси. По-долу е дадено подробно описание на предназначението на полетата VPI и VCI
PTI (Payload Type Identifier) - включва 3 бита и се използва за дефиниране на типа на данните.
CLP (Congestion Loss Priority) - с размер един бит, който задава приоритета на загуба при претоварване. Състоянието на този бит определя дали клетката ще бъде изхвърлена, ако се получи претоварване при нейното преминаване през мрежата.
HEC (Header Error Control) - с размер 8 бита и съдържа сума за контролиране на грешките, които могат да възникнат при предаване. Тази сума се изчислява само за заглавната част

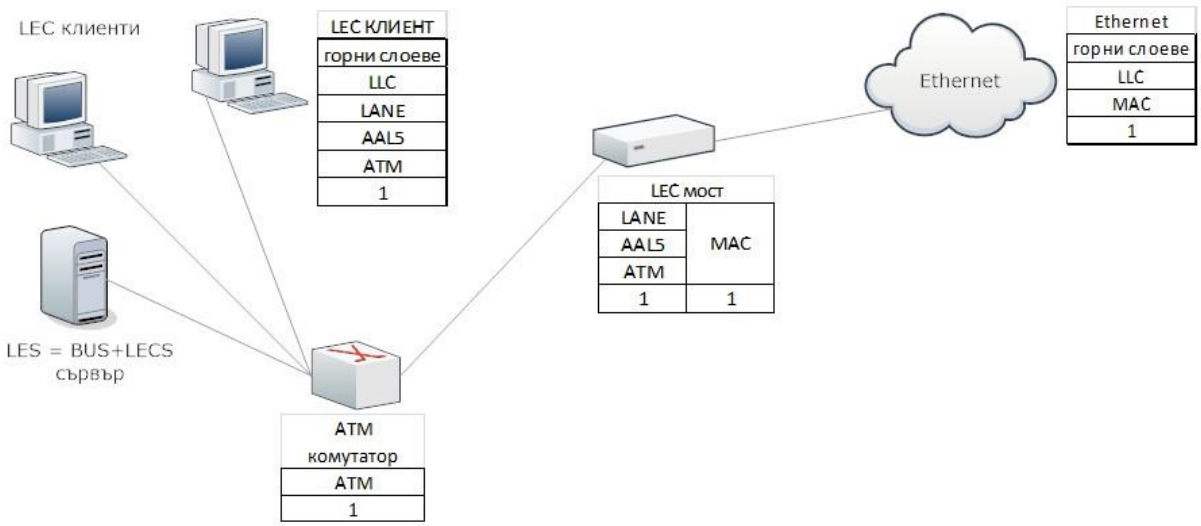
Заглавната част на клетки NNI не съдържа поле GFC. Вместо него полето VPI заема първите 12 бита на заглавието. Това позволява ATM

комутаторите да използват голям брой идентификатори на виртуален път. Останалите полета на заглавието NNI са идентични с тези на заглавието UNI.

ATM мрежите се различават от IEEE мрежите по това, че:

- ATM мрежите използват логически съединения, докато LAN предава данните без установяване на такова съединение;
- Мрежите Ethernet и Token Ring могат да използват broadcast и multicast предаване;
- ATM използва 28 битов VPI/VCI адрес, а споменатите локални мрежи – 48 битов MAC адрес.

За преодоляване на тези различия ATM форумът е разработил емулатор на локалните стандарти – LANE (LAN Emulation). Реализира се като драйвер (LEC-LINE Client). Такъв клиент трябва да бъде инсталиран на всеки компютър, свързан директно към ATM комутатор като го превръща в LEC клиент. Това е валидно и за всяко междинно устройство (маршрутизатор или мост) свързващо локална мрежа към ATM мрежа. Типа на междинното устройство се определя от това дали LAN мрежите, които свързва са еднотипни или не. При еднотипни се използва мост, а при разнотипни маршрутизатор. При LANE ключова роля изпълнява услугата LES (LAN Emulation Service), която включва два сървъра: BUS (Broadcast and Unknown Server) и LECS (LINE Configuration Server). Тя може да бъде стартирана в комутатор или крайно устройство. BUS сървърът обслужва broadcast и multicast съобщенията между LEC клиентите. LECS сървърът връща към LEC (при първоначалното му включване) адресите на LES и BUS сървърите, уточнява вида на локалната мрежа.



фигура 128 *LAN емулация*

На фигура 128 е изобразена примерна топология за LANE, като са показани и нивата спрямо OSI модела, до които функционират отделните устройства.

7.4.ISDN стандарт

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=28**

ISDN (Integrated Services Digital Network) дефинира цифрова мрежа с интегрирани услуги (1984 г.), предназначена за подмяна на старите POST системи. Проектирана е за пренос на глас и данни. Използва технологията комутиране на канали. Един ISDN комутатор (цифрова автоматична телефонна централа с добавени ISDN модули) дава възможност за достъп на абоната до следните услуги: некомутеруеми услуги от точка до точка, услуги с комутация на канали, услуги с комутация на пакети и услуги по управление на повикването. Каналите, които се използват при тази технология са:

- B-канал (bearer) с пропускателна способност от 64 Kbps, които могат да бъдат обединявани (инверсно мултиплексиране), за постигане на високоскоростни връзки.
- D-канал с пропускателна способност от 16 Kbps или 64 Kbps.

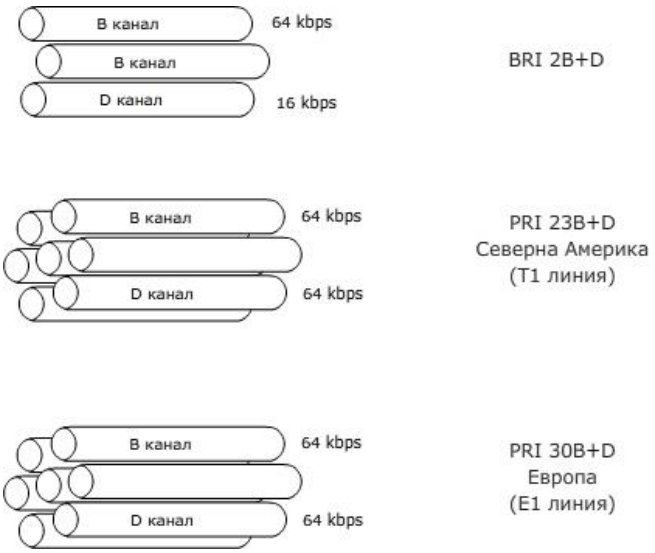
ISDN дефинира два вида абонатен интерфейс:

- BRI (Basic Rate Interface) – базов;
- PRI (Primary Rate Interface) – първичен.

BRI се предлага като 2B+D – предоставя на абонатите два канала от тип B и един от тип D (16 Kbps). При този интерфейс даден ISDN абонат може да използва единия B канал за предаване на данни, докато разговаря по телефона по другия B канал. D-каналът служи за предаване на служебна информация, която може да бъде сигнал за повикване, номерата на двата абоната, извеждане на информация за номера.

PRI за Европа се предлага като 30B+D – предоставя 30 B-канали и един D-канал със скорост 64 Kbps.

На фигура 129 са визуализирани коментираните интерфейси.



фигура 129 *BRI и PRI канали*

Третата стъпка може да запознае обучаемите с необходимото Оборудването може да се класифицира по следния начин:

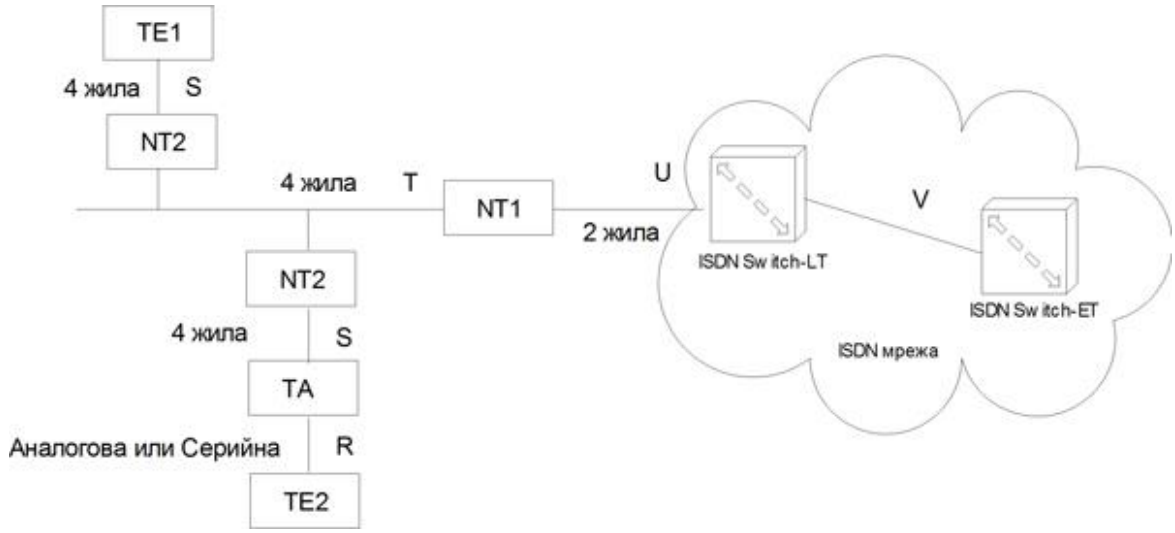
- Terminal Equipment (TE) с две разновидности:

- Terminal Equipment Type 1 (TE1) – ISDN терминали;
- Terminal Equipment Type 2 (TE2) – други терминали. За приобщаването им към ISDN мрежата се използват терминални адаптери (TA – Terminal Adapter).

- Network Termination 1 (NT1), Network Termination 2 (NT2) – крайни мрежови устройства. NT1 е устройство от физическо ниво свързващо T с U интерфейсите (фигура 130). NT2 е устройство, поддържащо до трето ниво от OSI модела;

- Line Termination (LN) – крайно линейно устройство. Реализира физическото ниво от OSI за връзката „абонатна линия - вход на ISDN комутатор“;
- Exchange Termination (ET) – крайно устройство на ISDN комутатор, реализиращо функциите до трето ниво включително;
- R интерфейс – между TE2 и TA;
- S интерфейс – между TE1 или TA и NT2;
- T интерфейс – между NT1 и NT2;
- U интерфейс – между NT1 и LT;
- V интерфейс – между LT и ET.

Позицията на всяко от тях е обозначена на фигура 130.



фигура 130. *Разположение на устройствата и интерфейсите в ISDN*

Отнасянето на стандарта към OSI модела е показано в таблица 11, където е представено разположението на протоколите спрямо различните нива.

OSI слоеве	D канал	B канал
мрежов	Q.931 (управление на повикването)	IP/IPX/AppleTalk/X.25
канален	LAPD (Q.921)	Frame Relay/LAPB/HDLC/PPP
физически	I.430 (BRI)/I.431 (PRI)/ANSI T1.601	

таблица 11 *Разположение на протоколите по нива при връзката „потребител-мрежа“*

От таблицата се вижда, че в каналния и мрежовия слой се използват различни протоколи в зависимост от вида на използвания канал и вида на

предоставяната услуга. Във физическия слой се използват едни и същи протоколи за двата вида канали.

За управление на D канала е дефиниран канален протокол LAPD (Link Access Protocol, D channel), който е част от семейството LAP протоколи, базирани на протокола HDLC. Форматът на кадъра за протокола LAPD има структура показана на фигура 131.

фигура 131. *Структура на LAPD кадър*

Service Access Point Identifier (SAPI) – идентифицира точката за достъп до услугите на протокола, обслужващи Ниво 3 (протокол Q.931).
First Address Extension bit (EA1) – заема стойност 0.
Command/Response bit (C/R) – задава типа на кадъра, команда или отговор.
Second Address Extension bit (EA2) - заема стойност 1.
Terminal Endpoint Identifier (TEI) – заема дефинирани стойности, свързани с използваното оборудване.

7.5.Стандарт B-ISDN

Широколентовият ISDN (B-ISDN - Broadband ISDN) е технология, предоставяща услуги, използващи скорости на предаване по-големи от тези на обикновения ISDN. B-ISDN се базира на оптика до абоната, ATM-мрежи и бърза комутация на пакети, докато ISDN на IDN и комутация на канали. Поддържаната скорости на предаване е до 622 Mbps. B-ISDN се състои от абонатно оборудване и множество междинни възли (B-ISDN централи), подобно на ISDN. От гледна точка на скоростта B-ISDN предлага следните услуги: пълнодуплексна (155.52 Mbps или 622.08 Mbps) и асиметрична (155.52 Mbps от абоната към мрежата и 622.08 Mbps в обратна посока).

7.6.Други мрежи с комутиране на канали

PSTN (аналоговите телефонни централи) са били част от нашето ежедневие, заедно с dialup връзките, които ни осигуряваха интернет чрез модем с максимална скорост от 56 Kbps.

DSL технологията осигурява WAN свързаност на базата на съществуващите медни проводници. Тя е финансово по-изгодна от други технологии и осигурява по-висока скорост. Една от разновидностите и масово се използва и на нашия пазар-ADSL. DSL може да бъде представена така:

- Постоянно включена технология;
- Предлага по-високи скорости от скъпоструващи връзки (напр. наети линии);
- По линията могат да бъдат предавани едновременно данни и глас.

В таблица 12за сравнение са показани някои от разновидностите на тази технология.

xDSL	Скорост на предаване		Характеристики
	към абоната (downstream)	от абоната (upstream)	
ADSL	1.544-12 Mbps	0.5-1.8 Mbps	Асиметричен поток в двете посоки; Разстояние от централата до 5 км.
ADSL2	1.544-12.0 Mbps	0.5-3.5 Mbps	
ADSL2+	24.0 Mbps	1.1-3.3 Mbps	
HDSL	0.784-2.0 Mbps	0.784-2.0 Mbps	Симетричен поток в двете посоки; Разстояние от централата до 3.8-20 км (с повторител); Използва T1/E1 преносни среди.
VDSL	до 52 Mbps	до 16 Mbps	Скъпа технология; Не е широко разпространена; Подходяща за аудио и видео.
VDSL2	до 100 Mbps	до 100 Mbps	
IDSL	до 144 Kbps	до 144 Kbps	По-скъпа и по- бавна от ADSL; Използва се при липса на възможности за други типове DSL връзки.

таблица 12 *Видове xDSL технологии*

Т-носещите са специализирани връзки за осигуряване на високи скорости на предаване [71]. Аналогичният стандарт за Европа са Е-носещи (таблица 13).

	Скорост на предаване	Брой канали (64Kbps)
Т-носеща(USA)		
Т-1	1.544 Mbps	24
Т-2	6.312 Mbps	96
Т-3	44.736 Mbps	672
Т-4	274.176 Mbps	4032
Е-носеща(Европа)		
Е-1	2.048 Mbps	30
Е-2	8.448 Mbps	120
Е-3	34.368 Mbps	480
Е-4	139.264 Mbps	1920
Е-5	565.148 Mbps	7680

таблица 13 *Технологии за пренос*

При реализацията в двата края се използват CSU/DSU устройства за кодиране на данните.

SONET (Synchronous Optical Network) е висикоскоростна оптична технология, функционираща във физическия слой и служи за основа на други технологии (например, B-ISDN). Базира се на така наречените OC (optical carrier) стандарти, показани в таблица 14.

OC стандарт	Скорост на предаване
OC-1(базова скорост)	51.84 Mbps
OC-3	155.52 Mbps
OC-12	622.08 Mbps
OC-36	1.866 Gbps

таблица 14 *Непълна таблица на OC стандартите*

8. Междумрежови комуникации

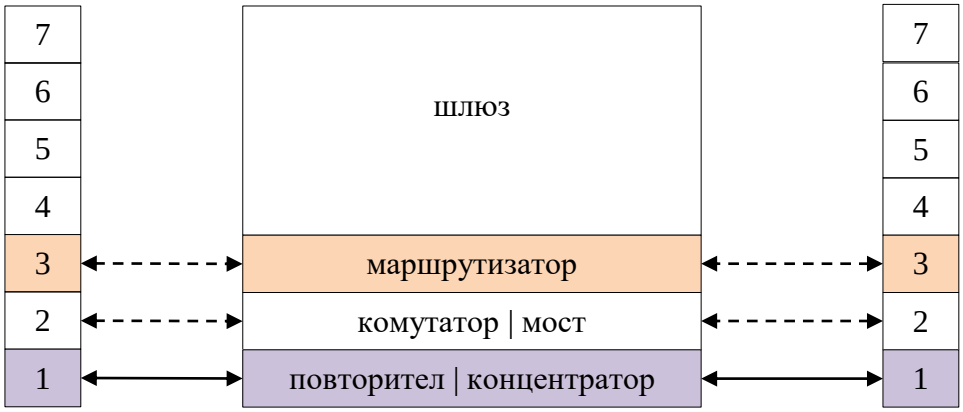
Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=30**

От представеното учебно съдържание дотук може да се заключи, че съществуват различни начини за изграждане на мрежова свързаност. Оттук произлиза и проблема за преодоляване на хетерогенността на мрежите при междумрежовите комуникации. Обект на внимание ще бъде съгласуването на две мрежи на ниво internetworking, което се извършва с помощта на допълнителни устройства поставени между тях. В зависимост от нивото за съгласуване се различават следните видове устройства: повторители, концентратори, модеми, мостове, комутатори, маршрутизатори, мост-маршрутизатори. Позицията, която заемат в общата схема, спрямо OSI модела, е демонстрирана чрез фигура 132.

8.1.Определения

Колизионен домейн – физически сегмент, където могат да възникнат колизии.

Броадкаст домейн – логически сегмент, определящ границите на броадкаст и мултикаст предаването.



фигура 132. *Разположение на устройствата спрямо OSI модела*

8.2.Повторител (Repeater)

Това са хардуерно устройство, работещо на първо ниво на OSI модела. Основните му функции са:

- възстановяване и усилване на сигнала;
- удължаване на покривното разстояние;

- съгласуване между сегментите във физическия слой.

8.3.Концентратор (hub)

Това е хардуерно устройство, работещо на първо ниво на OSI модела, което осигурява допълнителни входни точки за включване на устройства към локалната мрежа. Основната му функция е свързана с разпръскване на сигнала, постъпил на конкретен входен порт, до всички останали изходни портове. Концентраторите могат да бъдат:

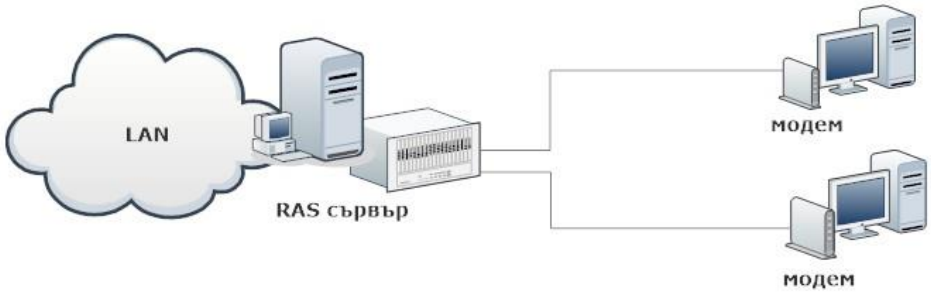
- пасивни – осигуряващи допълнителни входни точки;
- активни – допълнително усилващи сигнала за преодоляване на затихването му, с цел преодоляване на по-голямо разстояние;
- интелигентни – разполагат с процесор за диагностика на нефункциониращи портове.

Схемата на използване зависи от използвания стандарт. Например, при 10BaseT концентраторите могат да бъдат организирани до три нива в йерархична структура.

Всички портове на хъба принадлежат на един колизионен и един броадкаст домейн. Използва полудуплекс за предаване.

8.4.Модем (модулатор/демодулатор)

Модемът е специализирано комуникационно устройство за предаване на цифрови данни по аналогови (телефонни) трасета. Използват се за отдалечен достъп на единични потребители до локална или глобална мрежа. При достъп до локална мрежа се използва RAS-сървър (Remote Access Server) (фигура 133).



фигура 133 *Отдалечен достъп в локална мрежа*

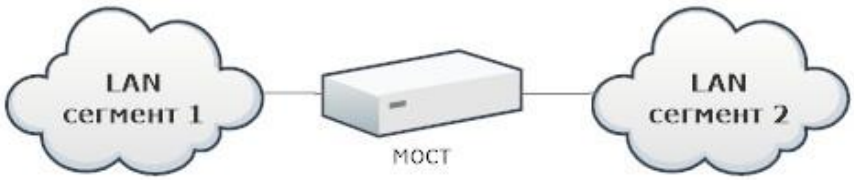
Различните типове модулация, които използват модемите са описани в т.2. В зависимост от разположението си те могат да се класифицират като:

- външни – отделно устройство, включено към един от серийните портове на компютъра;
- вътрешни – поставят се в слот на компютъра.

Скоростта, с която модемът може да предава информация, се измерва с единицата “бит в секунда”. Това е информационната скорост. Съществува и скорост на модулация. Тя се измерва в бодове и отчита изменението на състоянието на линията. Двете скорости могат и да съвпаднат, в някои от случаите.

8.5.Мост (bridge)

Мостът обслужва първи и втори слой на OSI модела чрез хардуерни и софтуерни реализации. Основно му предназначение е препредаване и филтриране на кадри, базирано на MAC адресите заложени в тях. Използват се най-често за сегментиране на големи и претоварени локални мрежи. Един мост разделя мрежата на два сегмента (фигура 134) като запазва MAC адресите на компютрите в таблица, указваща тяхната принадлежност. При предаване на кадър в даден сегмент мостът преценява дали да го препредаде на другия сегмент. Ако източникът и получателят са в един и същи сегмент това не се случва. Броадкаст кадрите (адресирани към FF-FF-FF-FF-FF-FF) и кадрите с адрес на получателя, неизвестен за маршрутната таблица на моста, се препращат и в двата сегмента.



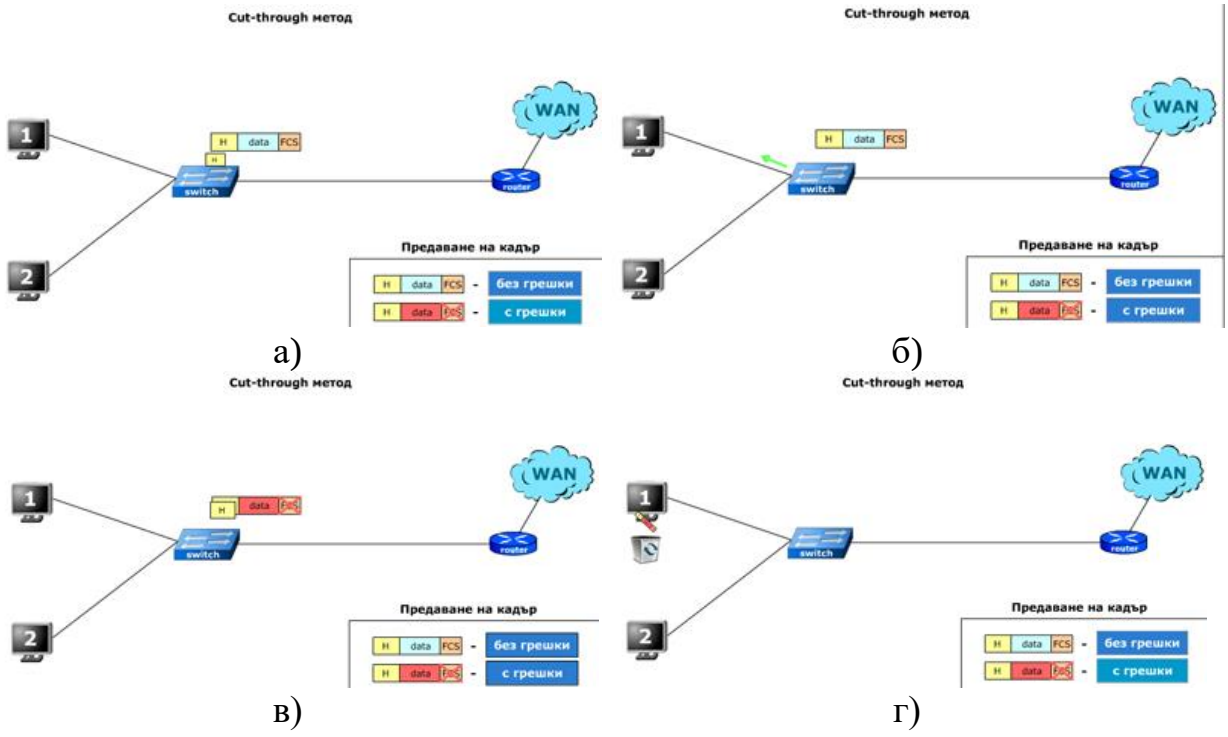
фигура 134 *Мост*

8.6.Комутатор (switch)

Комутаторът е устройство, комутиращо кадрите в каналния слой. Той разширява честотната лента, намалява колизиите и увеличава производителността на мрежата. Всеки негов порт сформира собствен колизионен домейн. Всички портове принадлежат на един броадкаст домейн. Препредава броадкаст и мултикаст съобщенията на всички портове с изключение на този от където идват. Комутаторът използва пълен дуплекс при предаване.

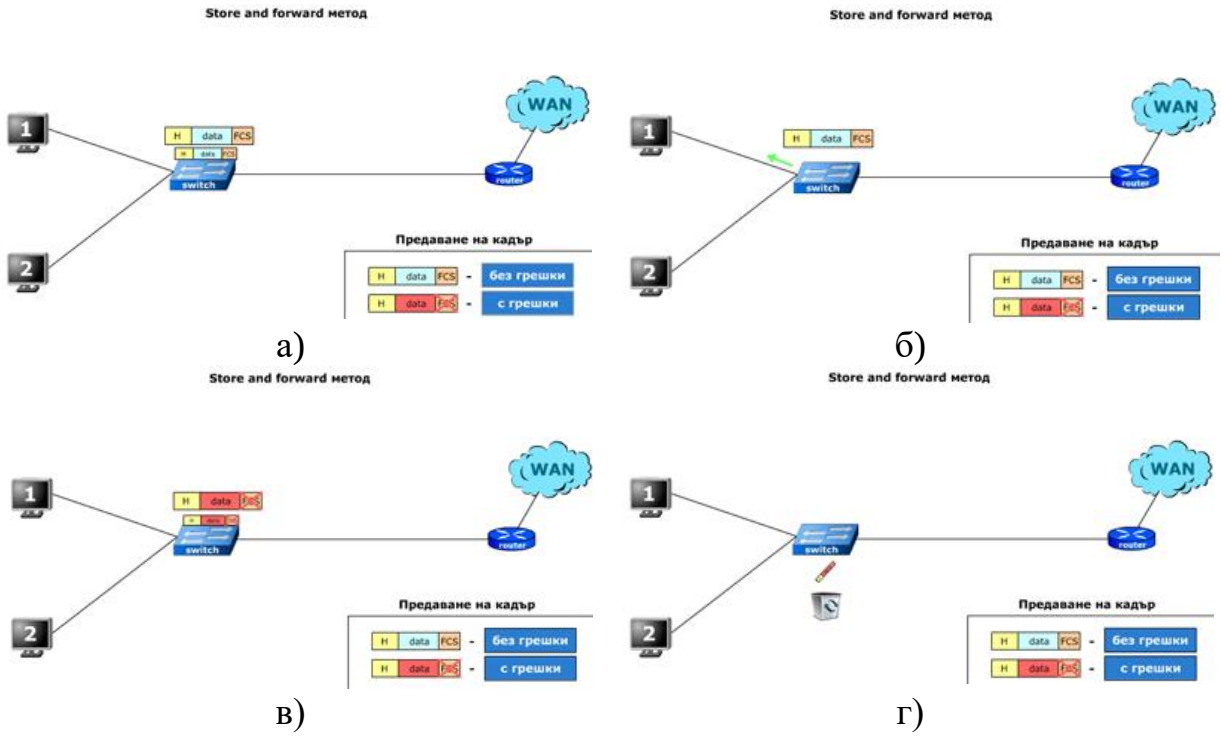
Съществуват три основни метода на комутация, които се използват от комутатора [21]:

- *Cut-through* –хедърът на пакета се запомня в буфера на порта. Чете се MAC адреса на дестинацията, който е в началото на кадъра след преамбюла. Определя се по адресната таблица порта за дестинация и резултата е незабавно препращане към целевия възел, въпреки продължаващото приемане на останалата част от кадъра. Откриването на грешки се извършва от получателя. Методът е създаден, за да намали забавянето в обработката на кадъра (латентния период в суича). На фигура 135 е демонстриран вариант за предаване на кадър без грешки (фигура 135а и фигура 135б) и с грешка (фигура 135в и фигура 135г).



фигура 135. *Cut-through метод*

- *Store and forward* – целият пакет се записва в буфера и се проверява за грешки. При правилно приемане на кадъра се предприема препредаването му към получателя (фигура 136а и фигура 136б). При грешка кадърът се унищожава (фигура 136в и фигура 136г). Този метод има по-голям латентен период на суича.



фигура 136. *Store and forwardметод*

- *Fragment-free* – хибриден метод между Cut-through и Store and forward, но запомня първите 64 байта преди да започне препредаване. Причината е, че грешките при комуникацията възникват най-често в този момент. Създаден за избягване на късни колизии.

Друга технология свързана с функционирането на суич е Transparent Bridging. Тази технология позволява на суича да получава необходимата информация за отделните възли, участващи в локалната мрежа без да е необходима намеса на администратор.

Според Cisco [21] при Transparent Bridging могат да се разграничат пет логически обособени части: обучение (Learning), запитване (Flooding), филтриране (Filtering), пренасочване (Forwarding), стареене (Aging).

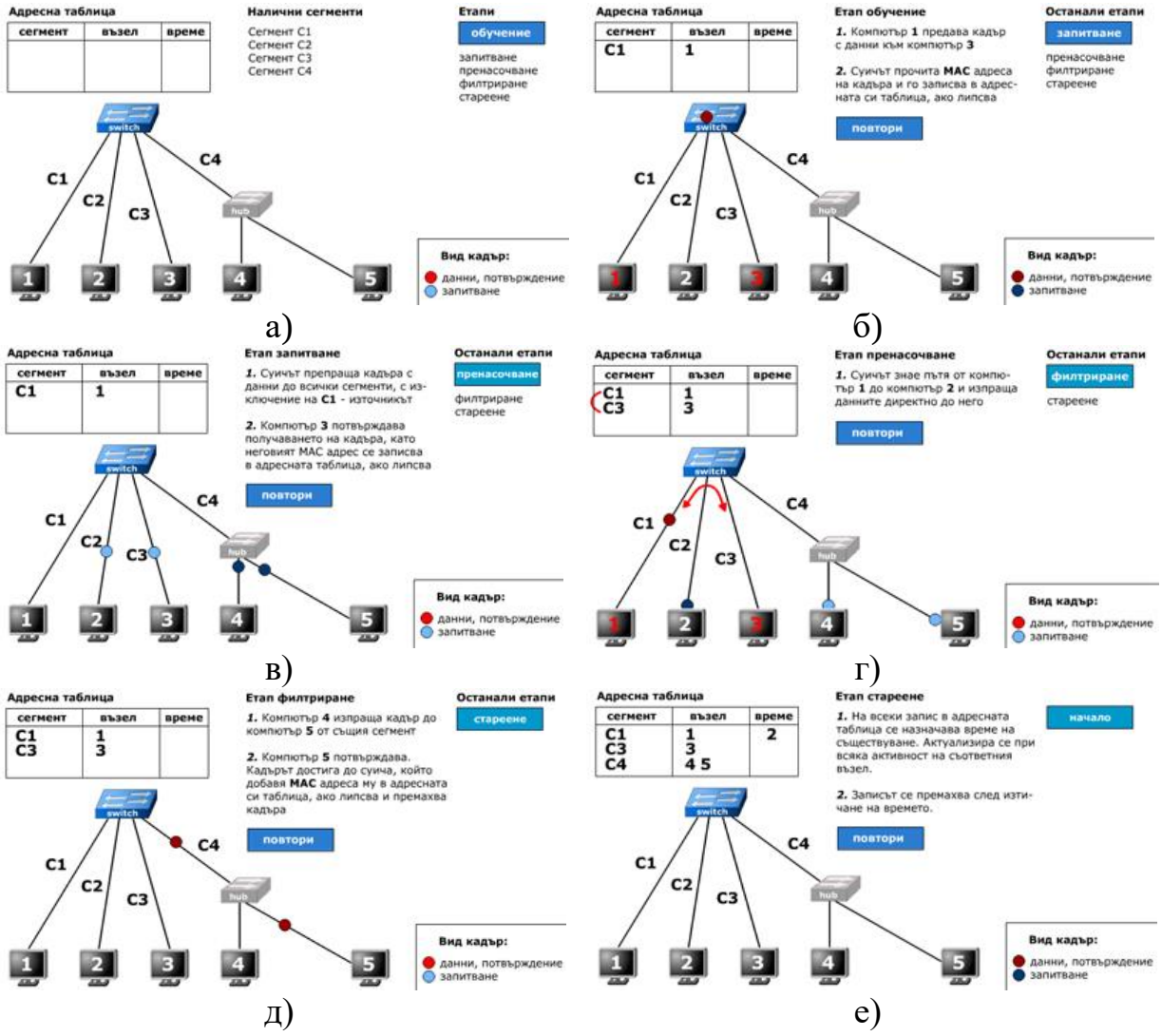
На фигура 137 са илюстрирани посочените етапи. В топологията на локалната мрежа като междинни устройства присъстват хъб и суич. Целта е нагледно съпоставяне на функционирането на двете устройства, както и показването на разновидност на сегменти, обособени от портовете на суича.

Адресната таблица на суича включва полета за сегмент, възел и време за съществуване на реда.

Налични са пет хоста номерирани в последователност от 1 до 5. Използването на този тип номерация е с цел избягване на дългото изписване на 6-байтовите MAC адреси на мрежовите адаптери. Обособени са четири сегмента с номерация **C1**, **C2**, **C3** и **C4**. Принадлежността на конкретен компютър към даден сегмент е указана в таблица 15.

Сегмент	Възел
C1	1
C2	2
C3	3
C4	4, 5

таблица 15 *Принадлежност на компютър към сегмент*



фигура 137. *Transparent Bridging*

Етап на обучение (фигура 137а и фигура 137б) – компютър 1 от сегмент C1 предава кадър с данни към компютър 3 от сегмент C3.

Преминаването на кадъра през суича предизвиква добавянето на MAC адреса на подателя към адресната му таблица, ако липсва.

Етап на запитване (фигура 137в) – ако суичът не знае къде се намира компютър 3 (получателят) изпраща запитване до всички сегменти различни от сегмента източник.

Компютър 3 потвърждава с кадър, предназначен за изпращача – компютър 1. Това дава възможност адресната таблица на суича да се обогати с още един MAC адрес и принадлежността му към конкретен сегмент, ако липсва.

Етап на пренасочване (фигура 137г) – всяка следваща комуникация между регистрирани в адресната таблица възли протича директно, без натоварване на останалите сегменти.

Етап на филтриране (фигура 137д) – компютър 4 изпраща кадър до компютър 5 от същия сегмент С4. Хъбът разпраща сигнала до останалите възли, свързани към него. Суичът приема кадъра и добавя компютър 4 към адресната си таблица, след което поради непознаване все още на компютър 5 разпраща запитване към останалите сегменти. В същото време компютър 5 потвърждава приемането на кадъра, изпратен към него с кадър, който достига обратно до компютър 4 и до суича. Суичът добавя и неговия адрес към сегмент С4 в таблицата си, след което игнорира кадъра. Всяка следваща комуникация между двата крайни възела ще протича само в сегмент С4, а кадрите от тази комуникация, достигнали до суича ще бъдат игнорирани.

Етап на стареене (фигура 137е) – тази техника се използва за избягване на натрупване на стара информация в адресната таблица на суича и възпрепятстване на неговото правилно функциониране. Информацията за всеки възел включва време за стареене, което се опреснява при всеки приет негов кадър. При липса на активност от страна на възела, той се премахва от списъка.

Категоризацията на комутаторите може да се направи на базата на нивата от OSI модела. Освен стандартните суичове от Ниво 2 съществуват и реализации, поддържащи функционалността от Ниво 3 и Ниво 4. Например, един комутатор от Ниво 3 работи в мрежовия слой на OSI модела т.е изпълнява функциите на един стандартен маршрутизатор.

Предимството на този вариант в сравнение със стандартния маршрутизатор е допълнителното хардуерно реализиране на функционалността на Ниво 3 и Ниво 4, което осигурява по-бързо изпълнение на определените действия.

8.7.Маршрутизатор (router)

Маршрутизаторът се използва за свързване на хетерогенни мрежи на нивото на мрежовия слой на OSI-модела. Участва като междинно устройство при изграждането на подмрежата. Разделя броадкаст домейните. Разполага с поне два мрежови интерфейса. Адресът на всеки от принадлежащите му интерфейси се нарича подразбиращ се шлюз (default gateway) за включената към него подмрежа. Маршрутизаторът извършва маршрутизиране на пакети, на базата на поддържаната от него рутираща таблица. Друга негова възможност е свързана с филтрирането на броадкаст кадри и пакети т.е. не позволява да се разпространяват извън конкретната подмрежа, което предотвратява т.нар. броадкаст буря (broadcast storm).

Използваните алгоритми за процеса на маршрутизация могат да бъдат:

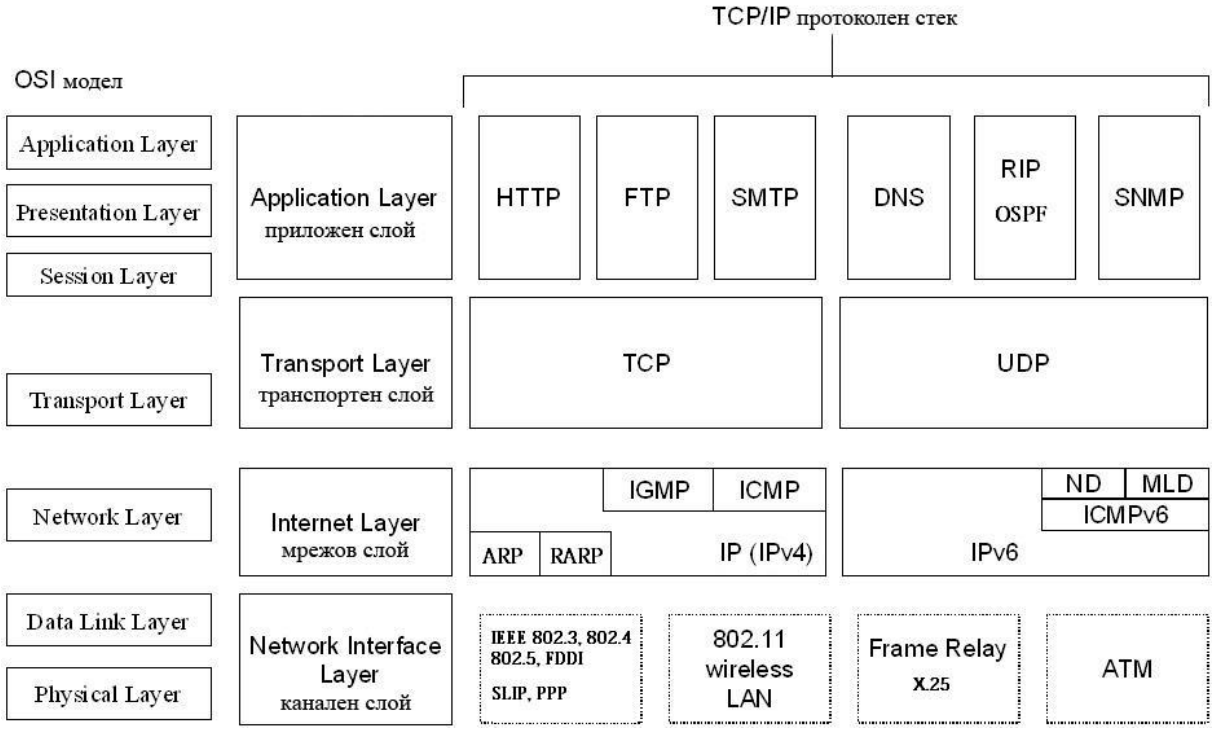
- статични – при тях администраторът поддържа и актуализира маршрутизиращите таблици на междинните устройства;
- динамични – междинните устройства използват маршрутизиращи протоколи за автоматично поддържане и актуализиране на техните таблици. Такива протоколи са RIP (Routing Information Protocol) и OSPF (Open Shortest Path First).

Не всички протоколи допускат маршрутизиране. Например, IPX, IP, XNS, DDP допускат маршрутизиране, докато NetBEUI (Microsoft), LAT (DEC) не допускат. Това се определя от използваната схема на адресиране от всеки от тях.

9. Комуникационен модел TCP/IP

Адрес: **http://kmk.fmi-plovdiv.org/kmk-lectures/mod/page/view.php?id=31**

TCP/IP (*Transmission Control Protocol/Internet Protocol*) е основния протоколен стек необходим за функционирането на глобалната мрежа Интернет (фигура 138). Това е основната причина за неговото по-подробно изучаване. Днес TCP/IP се поддържа стандартно от всички модерни операционни системи (UNIX, Windows NT, Windows 98, NetWare и др.).



фигура 138. *Слоеве и основни протоколи на TCP/IP*

Наименованието на стека се базира на двата основни протокола в него TCP и IP. Състои се от 4 слоя. На фигура 138 е показано отнасянето към OSI модела, основни протоколи и тяхното разположение по нива.

9.1.Канален слой

Каналният слой на модела TCP/IP съответстващ на първите два слоя на OSI модела, поддържа всички популярни стандарти за LAN (IEEE 802.3, 802.4, 802.5, FDDI), WAN (X.25, Frame Relay, ATM), SLIP и PPP.

Адресацията на това нови се определя от технологията, която се използва. При LAN това е MAC-адресът на мрежовия адаптер или порт. Всеки такъв адрес е уникален и се назначава от фирмата производител.

Заложен е хардуерно и всяка смяна на такъв тип устройство води до смяна и на MAC-адреса. Съставен е от 6 байта.

Пример: 00-24-D2-B5-73-04

Старшите 3 байта представляват идентификатор на фирмата производител, а младшите 3 байта идентифицират продукта и се назначават от самата фирма.

MAC-адресът уникално идентифицира всеки хост в локалната мрежа, като по този начин гарантира правилното и функциониране. Предаваните кадри в каналния слой се разпределят между хостовете благодарение на разпознати MAC-адреси.

9.2.Мрежов слой

Мрежовият слой се нарича още слой на междумрежовото взаимодействие. Неговата задача е да даде възможност на крайните възли да прехвърлят пакети с данни през произволни мрежа, като ги и маршрутизира.

Адресацията използва уникални IP адреси. Идентифицира хост или порт на междинен възел от мрежата. Ако разгледаме действащия стандарт IPv4 IP адресът е 4 байтов и се записват с 4 десетични числа в интервала от 0 до 255, разделени с точки.

Пример: 192.168.1.1

Логически се разделя на две части: адрес на мрежата (Net ID) и адрес на хоста (Host ID). Ролята на разделител изпълнява т.нар. мрежова маска, която е 4 байтова и гарантира гъвкавост при определянето на границите.

Пример: 255.255.255.0

Мрежовата маска винаги придружава IP адрес. Изискването към нея, в двоичен вид, е да започва и да продължи с единици до приключване на последователността им, след което останалите позиции се запълват с двоични нули. Не се допуска смесване на 0 и 1. Маската участва в операцията двоично умножение с IP адрес и помага за определяне на мрежовия сегмент, в който се намира устройството.

На базата на този критерии може да бъде определяна валидността на конкретна мрежова маска.

Пример:

маска	двоичен вид	статус
255.255.255.0	11111111.11111111.11111111.00000000	валидна
255.128.0.0	11111111.10000000.00000000.00000000	валидна
255.64.0.0	11111111.**01**000000.00000000.00000000	невалидна

таблица 16 *Валидност на мрежова маска*

255.64.0.0 е невалидна мрежова маска, защото 64 представено в двоичен вид позволява присъствието на 0 в най-старшия си бит, което нарушава условието за непрекъснатост на последователността от 1 (таблица 16).

Съществуват 5 класа IP адреси (по IPv4) [49]. Областта им на действие ги разделя на публични и частни. Публичните адреси са валидни за цялата IP мрежа, където функционират, а частните могат да бъдат назначавани само на хостове от локална мрежа (компютър с такъв адрес не може да бъде представян директно в Интернет).

Наименованието на отделните класове и диапазона на стойността на първия байт от адреса в десетичен вид са показани в таблица 17.

Клас	Десетична стойност на първия байт
клас A	от 1 до 126 включително
клас B	от 128 до 191 включително
клас C	от 192 до 223 включително
клас D	от 224 до 239 включително
клас E	от 240 до 254 включително

таблица 17 *Диапазон на десетичните стойности на първия байт*

Като особеност трябва да се отбележи, че:

- десетичната нула не се използва като първо число на IP-адрес;
- адреси, започващи с числото 127 се използва само за специални цели. Например, 127.0.0.1 се използва за обратна връзка (loopback), т.е. генерираните от възела данни се предават обратно към горните слоеве все едно, че току-що са приети от мрежата;
- мрежовият адрес се отличава от адрес на хост по това, че всички позиции заделени за HostID са запълнени с двоични 0.

Двоичната комбинация, с която започва първият байт на адреса е показана в таблица 18.

Клас	Двоична комбинация за начало на първия байт	
клас A	0xxxxxxx	гарантира интервал [1,127]
клас B	10xxxxxx	гарантира интервал [128,191]
клас C	110xxxxx	гарантира интервал [192,223]
клас D	1110xxxx	гарантира интервал [224,239]
клас E	11110xxx	гарантира интервал [240,255]

таблица 18 *Двоична комбинация за начало на първия байт*

Шаблоните указват позициите от битове в IP адреса, принадлежащи на адреса на мрежата и адреса на хоста. Шаблонът за конкретния клас адреси съдържа позиции означени с n, съответстващи на последователността от битове, принадлежащи на Net ID, h-позиции за битовете, принадлежащи на Host ID и x-позиции за неангажирани битове (таблица 19).

Клас	Шаблон
клас A	0nnnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh
клас B	10nnnnnn.nnnnnnnn.hhhhhhhh.hhhhhhhh
клас C	110nnnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh
клас D	1110xxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx
клас E	11110xxx.xxxxxxxx.xxxxxxxx.xxxxxxxx

таблица 19 *Шаблони*

Стандартните мрежови маски за клас могат да се видят в таблица 20.

Клас	Мрежова маска
клас A	255.0.0.0
клас B	255.255.0.0
клас C	255.255.255.0
клас D	липсва
клас E	липсва

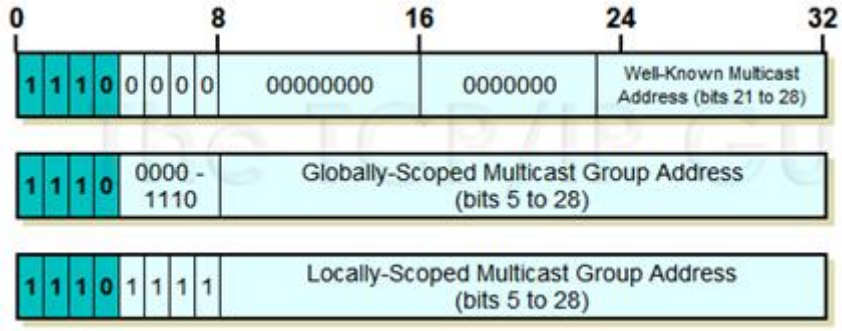
таблица 20 *Стандартни мрежови маски*

Относно адресите от клас D се използват за изпращане на multicast-съобщения до определена група хостове, на която е присвоен указаният адрес [25]. Multicast-адресът не се дели на Net ID и Host ID, което обяснява липсата на мрежова маска. Разновидности са показани в таблица 21.

Начало	Край	Описание
224.0.0.0	224.0.0.255	“well-known” multicast адреси – дефинирани са от IANA и са предназначени за локалните подмрежи. Например: RIP-224.0.0.9, OSPF-224.0.0.5 и 224.0.0.6
224.0.1.0	238.255.255.255	Globally-scoped (Internet) multicast адреси – могат да се назначават на различни групи в Интернет пространството. Например, Network Time Protocol-224.0.1.1

239.0.0.0	239.255.255.255	Administratively-scoped (local) multicast адреси – адресират групи на локално ниво

таблица 21 *Разновидности на мултикаст адресите*



фигура 139. *Multicast адреси*

Статистическата информация за броя на мрежите и адресите са представени в таблица 22.

Клас	Брой мрежи	Брой адреси
клас A	127	16777216
клас B	16384	65536
клас C	2097152	256

таблица 22 *Статистическа информация за поддържани мрежи и адреси*

За конфигуриране на локални мрежи, без да е необходимо заемането на валидни интернет адреси се използват частни такива. Дефинираните области са три:

- **клас A** - 10.0.0.0/8 - от 10.0.0.0 до 10.255.255.255 (24 битов блок от адреси);
- **клас B** - 172.16.0.0/12- от 172.16.0.0 до 172.31.255.255 (20 битов блок от адреси);
- **клас C** - 192.168.0.0/16 – от 192.168.0.0 до 192.168.255.255 (16 битов блок от адреси);

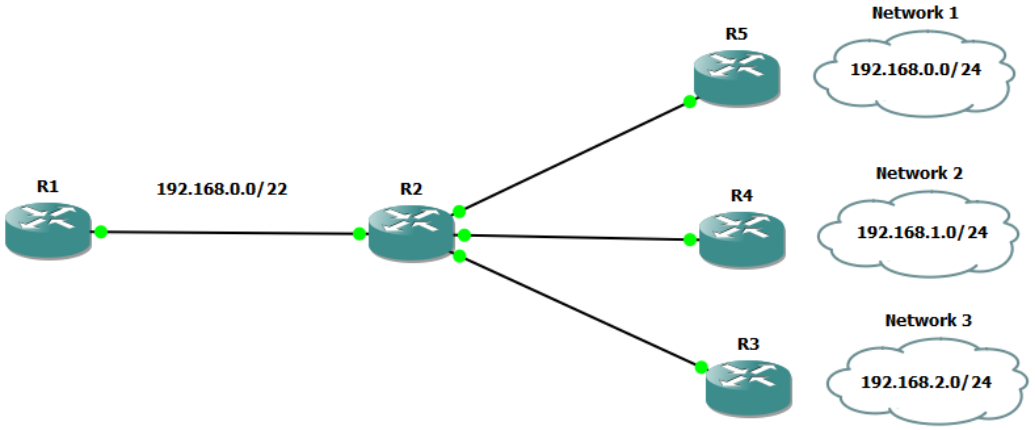
Определени са и Link-local адреси за случаи, при които хостът не може да получи TCP/IP конфигурационни параметри от DHCP сървър или не са му назначени статично. Хостът си определя адрес в обхвата 169.254.0.0 - 169.254.255.255. Подходът позволява отделни хостове с такива настройки да попаднат в една мрежа и да комуникират помежду си.

Означенията от вида 10.0.0.0/8 са от използваната алтернатива за безкласово адресиране, базирано на безкласовата междудомейнова

маршрутизация CIDR (classless inter-domain routing), позволяващо по-ефективно разпределение на IP адреси. Числото 8 след наклонената черта означава, че 8-те най-леви бита се използват за идентификация на мрежата (префикс), а останалите за разпознаване на хоста. Обикновенно префиксите варират между /13 и /27 и се отъждествяват с брой мрежи от клас C. Например, CIDR адресен блок с префикс:

- /27 се равнява на $\frac{1}{8}$ мрежи от клас C (32 хост адреса).
- /26 се равнява на $\frac{1}{4}$ мрежи от клас C (64 хост адреса).
- /25 се равнява на $\frac{1}{2}$ мрежи от клас C (128 хост адреса).
- /24 се равнява на 1 мрежа от клас C (256 хост адреса).
- /15 се равнява на 512 мрежи от клас C (131 072 хост адреса).

CIDR позволява подходите supernetting (използване на съседни блокове от адресното пространство за симулиране на по-голямо адресно пространство) и route aggregation (едно вписване в маршрутната таблицата на рутер, представляващо множество мрежи). Целта е намаляване на размера на този тип таблици.



фигура 140 *Supernetting и aggregation*

На фигура 140 са представени три последователни мрежи от клас C с префикс /24. В двоичен вид те изглеждат по следния начин:

IP	еднакви битове	разлика
192.168.0.0	11000000.10101000.000000	**00.00000000**
192.168.1.0	11000000.10101000.000000	**01.00000000**

192.168.2.0	11000000.10101000.000000	**10.00000000**

Вертикалната черта показва докъде достигат отляво-надясно еднаквите битове. Тяхната бройка формира новия префикс на супермрежата. Адресът на самата мрежа се получава като същите битове се допълват с нули до получаване на правилен мрежов адрес:

супермрежа	**нов префикс**	**еднакви битове**	**допълване**
192.168.0.0	**22**	11000000.10101000.000000	**00.00000000**

Третата стратегия, която според възникването си в исторически план се подрежда на второ място след класовата адресация и се наследява при безкласовата адресация за определяне на префикса, се нарича Variable-Length Subnet Masking (VLSM, RFC 950) [51]. При нея се използват подмаски (броя на единиците е различен от тези на стандартните маски) за дефиниране на подмрежи. Съществуват и някои ограничения, които отпадат при CIDR.

9.2.1. Задача за калкулиране на подмрежови маски, подмрежи, адреси на устройства от мрежата

Задачите на този етап на обучение могат да бъдат групирани в две основни групи:

Група 1. Задачи за определяне на валидността на мрежова маска

Примерна задача 1: Посочете невалидната мрежова маска: 255.255.255.252, 255.255.64.0, 255.255.192.0, 255.255.252.0.

Стъпки при решаване на задачата:

Стъпка 1: Определя се позицията на първия байт (отляво-надясно) от структурата на маската, чиято стойност е различна от 255. Ако неговата десетична стойност е 0 то следващите го байтове (ако има такива) трябва да бъдат също 0. Ако десетичната стойност на байта е между 0 и 255, то тя се превръща в двоичен вид и се преминава към *Стъпка 2*.

Първите байтове, отляво-надясно, между 0 и 255, в примерите, са подчертани:

маска	**двоично представяне**
255.255.255.**252**	11111111.11111111.11111111.**11111100**

255.255.**64**.0	11111111.11111111.**01000000**.00000000
255.255.**192**.0	11111111.11111111.**11000000**.00000000
255.255.**252**.0	11111111.11111111.**11111100**.00000000

Стъпка 2: В двоичния вид се проследява разпределението на битовете с единици. Ако последователността от единици започва от най-ляво и не се нарушава от нули до нейното завършване и останалите байтове надясно (ако има такива) са нули, то мрежовата маска е правилна. Останалите случаи генерират грешна мрежова маска.

От зададените по-горе примери, грешна последователност се съдържа само в предложението за маска-255.255.**64**.0. Грешката е в първата и втората позиция, отляво-надясно, в третия байт (64 представено в двоичен вид съдържа 0 в най-старшия си бит, с което се нарушава условието за непрекъснатост на последователните 1).

Група 2. Задачи за калкулиране на подмрежови маски, подмрежи, адреси на устройства от мрежата

Всеки бит от маската определя как да се тълкува съответният бит от IP адреса:

- нулевата стойност на бит от маската означава, че сътветният бит от IP адреса принадлежи към адреса на хоста;
- единица като стойност на бит от маската означава, че съответният бит от IP адреса принадлежи към адреса на мрежата.

Използваната формула за определяне на броя на необходимите подмрежи при VLSM стратегията е 2^n-2, където:

- n е броя на битовете от адреса на хоста, които трябва да се пречислят към адреса на подмрежата;
- -2 означава да се пропуснат подмрежи с номера, съдържащи само нули(Subnet Zero) или единици в областта за подмрежа.

При CIDR броят на подмрежите се постига с удължаване на мрежовия префикс. В този случай не е актуално избягването на споменатите гранични подмрежи, което означава, че горепосочената формула придобива вида 2^n.

Използването на всеки от изброените подходи зависи от възможностите на маршрутизиращите протоколи, заложени в маршрутизаторите. Примерни протоколи, поддържащи VLSM и CIDR са: Routing Information Protocol Version 2 (RIP2), Open Shortest Path First protocol (OSPF), Cisco EIGRP и др. Те обменят информация и за мрежовата подмаска (префикса), което помага на рутерите да вземат правилното решение при маршрутизирането.

Следващата задача показва разликата при използване на двата подхода, VLSM и CIDR, за определяне на броя на подмрежите. Терминът „мрежова маска“ се обвързва с VLSM стратегията, а терминът „префикс“ със CIDR.

Примерна задача 2: Каква мрежова маска (префикс) трябва да се постави, за да може мрежа с адрес 192.168.10.0 да се разбие най-малко на 3 подмрежи?

Стъпки при решаване на задачата за първи случай – използване на VLSM:

Стъпка 1: Определя се минималната стойност на n, за която изразът 2^n-2 е по-голям или равен на броя на цитираните подмрежи (в случая 3).

$$3 \leq 2^n-2 \Rightarrow n=3$$

Стъпка 2: Определя се стандартната мрежова маска за посочения адрес, в случая 255.255.255.0. Разписва се в двоичен вид.

Стъпка 3: Последователността от единици се допълва с нови 3 бита от най-старшите в четвъртия байт и се получава резултат 255.255.255.224.

тип маска		**двоичен вид**		
стандартна	255.255.255.0	11111111.11111111.11111111.	**000**	**00000**
VLSM	255.255.255.224	11111111.11111111.11111111.	**111**	**00000**

Стъпки при решаване на задачата за втори случай – използване на CIDR:

Стъпка 1: Определя се минималната стойност на n, за която изразът 2^n е по-голям или равен на броя на цитираните подмрежи (в случая 3).

$$3 \leq 2^n \Rightarrow n=2$$

Стъпка 2: Определя се стандартната мрежова маска за посочения адрес, в случая 255.255.255.0 (префикс /24). Разписва се в двоичен вид.

Стъпка 3: Последователността от единици се допълва с нови 2 бита от най-старшите в четвъртия байт и се получава резултат 255.255.255.192 (префикс /26).

тип маска (/префикс)		**двоичен вид**		
станд. (/24)	255.255.255.0	11111111.11111111.11111111.	**00**	**000000**
VLSM (/26)	255.255.255.192	11111111.11111111.11111111.	**11**	**000000**

Примерна задача 3: Кои са новополучените подмрежи, шлюзове и IP адреси за назначаване на мрежата *от Примерна задача 3* при използване на CIDR?

Стъпки при решаване на задачата:

Стъпка 1: Адресът на мрежата (192.168.10.0) и мрежовият префикс (/26) се представят в двоичен вид (може само четвъртият байт);

192.168.0.0	11000000.10101000.00001010.	**??**	**000000**
255.255.255.192	11111111.11111111.11111111.	**11**	**000000**

Стъпка 2: Определят се възможните комбинации за битовете от адреса:

00, **01**, **10**, **11**

Стъпка 3: Съставя се таблица 23 и се избират три от новосъздадените подмрежи като се спазват следните правила:

- Мрежов адрес – позициите на хоста са запълнени с нули;
- Broadcast адрес - позициите на хоста са запълнени с единици;
- Адрес на шлюза – първия адрес от тези за назначаване;

Номер	Описание	Двоично представяне	Десетичен вид
1	Мрежов адрес	11000000.10101000.00001010.**00**000000	192.168.10.0
	Първи IP адрес	11000000.10101000.00001010.**00**000001	192.168.10.1

	Последен IP адрес	11000000.10101000.00001010.**00**111110	192.168.10.62
	Broadcast адрес	11000000.10101000.00001010.**00**111111	192.168.10.63
2	Мрежов адрес	11000000.10101000.00001010.**01**000000	192.168.10.64
	Първи IP адрес	11000000.10101000.00001010.**01**000001	192.168.10.65
	Последен IP адрес	11000000.10101000.00001010.**01**111110	192.168.10.126
	Broadcast адрес	11000000.10101000.00001010.**01**111111	192.168.10.127
3	Мрежов адрес	11000000.10101000.00001010.**10**000000	192.168.10.128
	Първи IP адрес	11000000.10101000.00001010.**10**000001	192.168.10.129
	Последен IP адрес	11000000.10101000.00001010.**10**111110	192.168.10.190
	Broadcast адрес	11000000.10101000.00001010.**10**111111	192.168.10.191
4	Мрежов адрес	11000000.10101000.00001010.**11**000000	192.168.10.192
	Първи IP адрес	11000000.10101000.00001010.**11**000001	192.168.10.193
	Последен IP адрес	11000000.10101000.00001010.**11**111110	192.168.10.254
	Broadcast адрес	11000000.10101000.00001010.**11**111111	192.168.10.255

таблица 23 *Таблица с възможните подмрежи удовлетворяващи условието на задачата*

Примерна задача 4: Кои са новополучените подмрежи, шлюзове и IP адреси за назначаване на мрежата от задача 6.2 при използване на VLSM? Посочете отпадащите гранични подмрежи.

Стъпки при решаване на задачата:

Стъпка 1: Адресът на мрежата (192.168.10.0) и мрежовата маска (255.255.255.224) се представят в двоичен вид (може само четвъртият байт);

192.168.0.0	11000000.10101000.00001010.	**???**	**00000**
255.255.255.224	11111111.11111111.11111111.	**111**	**00000**

Стъпка 2: Определят се възможните комбинации за битовете от адреса:

000, **001**, **010**, **011**, **100**, **101**, **110**, **111**

Стъпка 3: Определят се отпадащите гранични подмрежи:

000, 111

Стъпка 4: Съставя се таблица 24 и се избират три от новосъздадените подмрежи:

Номер	Описание	Двоично представяне	Десетичен вид
1	Мрежов адрес	11000000.10101000.00001010.**001**00000	192.168.10.32
	Първи IP адрес	11000000.10101000.00001010.**001**00001	192.168.10.33
	Последен IP адрес	11000000.10101000.00001010.**001**11110	192.168.10.62
	Broadcast адрес	11000000.10101000.00001010.**001**11111	192.168.10.63
2	Мрежов адрес	11000000.10101000.00001010.**010**00000	192.168.10.64
	Първи IP адрес	11000000.10101000.00001010.**010**00001	192.168.10.65
	Последен IP адрес	11000000.10101000.00001010.**010**11110	192.168.10.94
	Broadcast адрес	11000000.10101000.00001010.**010**11111	192.168.10.95
3	Мрежов адрес	11000000.10101000.00001010.**011**000000	192.168.10.96
	Първи IP адрес	11000000.10101000.00001010.**011**000001	192.168.10.97
	Последен IP адрес	11000000.10101000.00001010.**011**111110	192.168.10.126
	Broadcast адрес	11000000.10101000.00001010.**011**111111	192.168.10.127
4	Мрежов адрес	11000000.10101000.00001010.**100**000000	192.168.10.128
	Първи IP адрес	11000000.10101000.00001010.**100**000001	192.168.10.129
	Последен IP адрес	11000000.10101000.00001010.**100**111110	192.168.10.158
	Broadcast адрес	11000000.10101000.00001010.**100**111111	192.168.10.159
5	Мрежов адрес	11000000.10101000.00001010.**101**000000	192.168.10.160
	Първи IP адрес	11000000.10101000.00001010.**101**000001	192.168.10.161
	Последен IP адрес	11000000.10101000.00001010.**101**111110	192.168.10.190
	Broadcast адрес	11000000.10101000.00001010.**110**111111	192.168.10.191
6	Мрежов адрес	11000000.10101000.00001010.**110**000000	192.168.10.192
	Първи IP адрес	11000000.10101000.00001010.**110**000001	192.168.10.193
	Последен IP адрес	11000000.10101000.00001010.**110**111110	192.168.10.222
	Broadcast адрес	11000000.10101000.00001010.**110**111111	192.168.10.223

таблица 24 *Таблица с възможните подмрежи удовлетворяващи условието на задачата*

9.2.2. NAT (Network address Translation)

NAT (Network address Translation) е подход, осигуряващ преобразуване на локални IP адреси в един или няколко глобални IP адреси. Целта е представяне на локалните хостове в Интернет пространството чрез ограничено използване на глобални IP адреси. Съществуват три основни реализации на този подход, поддържани от маршрутизиращите устройства:

- статична – съпоставя на всеки вътрешен адрес различен глобален адрес;
- динамична – обвързва група от локални адреси с един или няколко глобални адреси. За целта в NAT таблиците на маршрутизаторите, където се асоциират локален с глобален IP адрес, се добавя и порт, идентифициращ комуникационния процес;
- смесена – обединява изброените преди това реализации.

локален адрес	глобален адрес
192.168.0.5	87.97.197.51
192.168.0.15	87.97.197.52
...	...

таблица 25 *Статично разпределение*

локален адрес	порт	глобален адрес	порт
192.168.0.5	80	87.97.197.51	6880
192.168.0.15	3389	87.97.197.51	6689
...	...	...	...

таблица 26 *Динамично разпределение*

9.3. Транспортен слой

Транспортният слой има за цел да скрие от приложните процеси детайлите по осъществяване на връзката и предаването на данните. Двете важни понятия свързани с адресацията на това ниво са порт и сокет (socket).

Портовете се използват от протоколите TCP и UDP за връзка с приложните процеси. Портът представлява 16-битово число. Някои от портовете са стандартно резервирани (например порт 21 – за протокола FTP).

Портът и IP-адресът съвместно образуват сокет (socket), например: 89.68.180.5:21. Двойка сокети (от двете страни на комуникацията)

еднозначно идентифицира едно TCP-съединение. Един сокет може да участва в няколко съединения едновременно.

9.4.Приложен слой

Приложният слой осигурява набор от програми, всяка от които предоставя определен вид мрежова услуга. Основно средство за адресация на това ниво са DNS-имената на хостове. Те са част от разпределената база данни, наречена DNS (Domain Name System), поддържаща йерархична система от имена за идентифициране на възлите и ресурсите в Internet. Всяко име е изградено на йерархичен принцип – състои се от отделни части разделени с точка, като най-дясната част представлява име на област, която е най-високо в йерархията (нарича се още “top-level” домейн). Синтаксисът на DNS-имената е следния:

поддомейнN.поддомейнN-1. … .поддомейн1.домейн

Пример за DSN-име е името kmk.fmi-plovdiv.org, където kmk е поддомейн на домейна fmi-plovdiv.org, а org е “top-level” домейн.

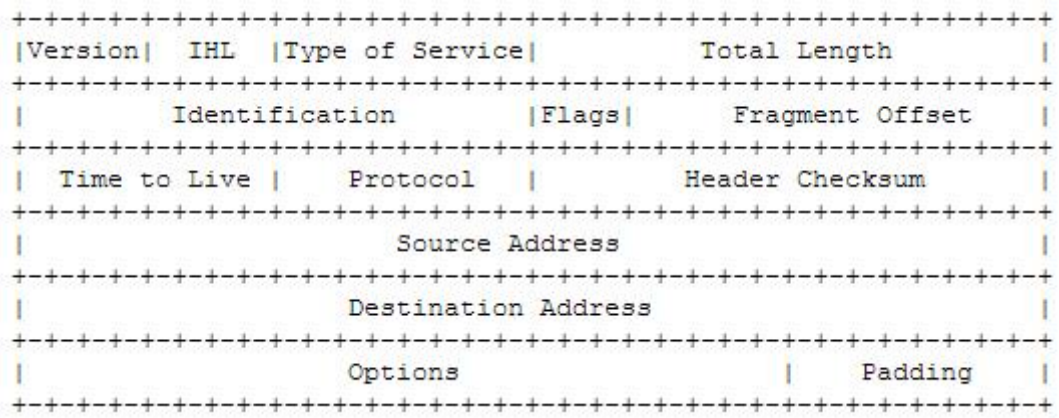
Основното предназначение на DNS е автоматично търсене на IP-адрес по съответното му DNS-име. За тази цел се използва протоколът DNS за приложния слой. Спецификацията на DNS се определя от документите RFC 1034 и RFC1035.

9.5.Протоколи

9.5.1. IP (Internet Protocol)

IP е основен протокол в TCP/IP. Осигурява използването на IP адреси и маршрутизирането на пакета от мрежата на подателя през междинните машрутизатори до мрежата на получателя. При необходимост големите пакети могат да бъдат фрагментирани, така че да отговарят на изискванията на подмрежата, през която преминават. IP счита всеки пакет за самостоятелна единица, не гарантира неговата доставка и не контролира потока от данни. Игнорира сгрешените пакети и използва ICMP съобщение за информиране на изпращача. Разчита на протоколите от по-горен слой за надеждното предаване на данните. IP е описан в документ RFC 791.

Структурата на заглавната част на пакета и описанието ѝ е показано на фигура 141.

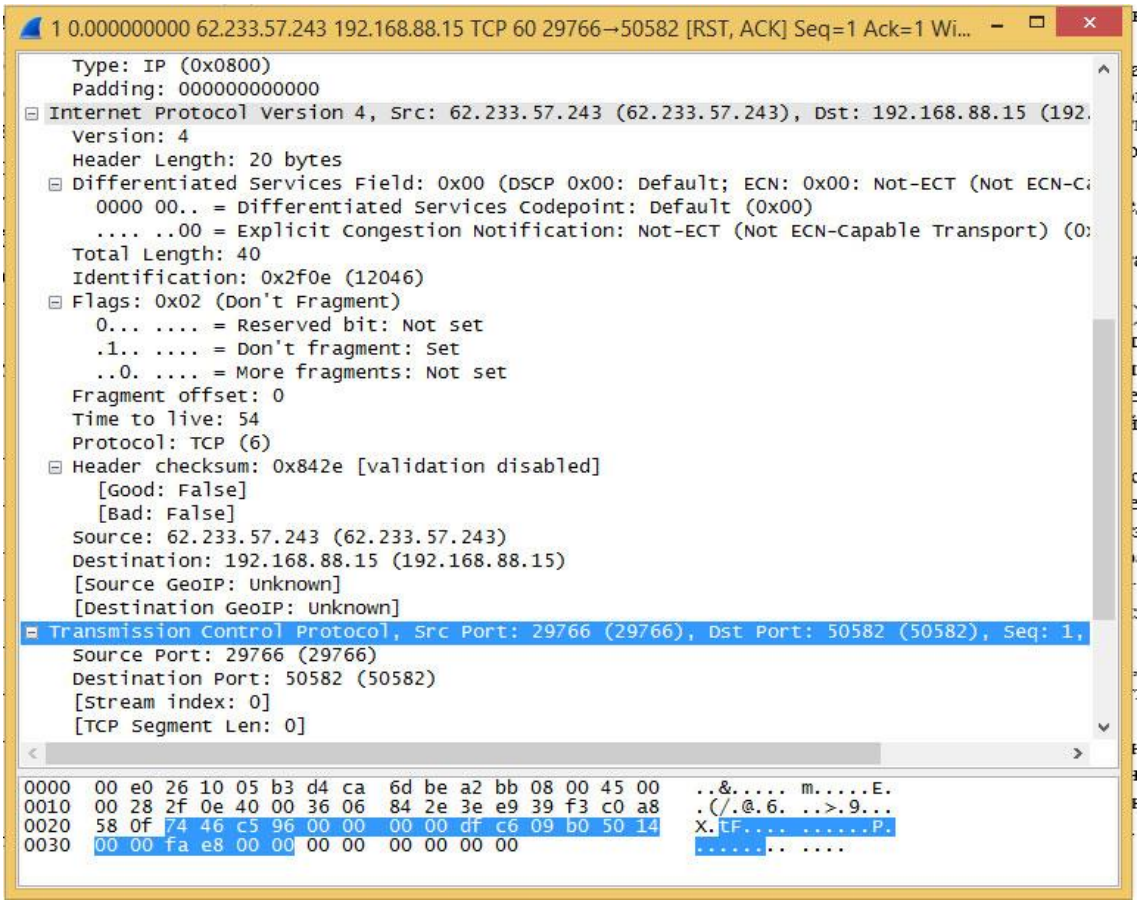


фигура 141 *Заглавна част на пакет*

Version – 4-битово поле, посочващо версията на протокола (IPv4-актуалната).
IHL (Internet Header Length) – 4-битово поле, задаващо дължината на IP хедъра в байтове. Реалната дължина се получава при умножение на заложеното число х 4 байта (минимум 20 байта т.е. 5х4 байта). Максималната стойност за полето е 15, което означава 15х4 байта=60 байта.
ToS (Type of Service) – 8-битово поле предназначено за маршрутизаторите при вземане на решение, относно пътя за преминаване на пакета. Заема предимно стойност 0. Някои маршрутизиращи протоколи като OSPF, EIGRP,IGRP, BGP разпознават тези битове, а други като RIP v1,v2 не ги разпознават. Първите 4 бита задават приоритета, а вторите четири типа на обслужването. Означенията за вторите са:
0000 – нормално обслужване;
0001- минимизация на разходите;
0010 – максимална надеждност;
0100 – максимална пропускателна способност;
1000 – минимално забавяне.
Total Length – 16-битово поле, задаващо дължината на IP пакета (65535 байта, от които между 20 и 60 байта са за хедър). Например, протоколът CSMA-CD налага ограничение на данните, които могат да се заложат в кадъра, от 46 до 1500 байта. По-малък кадър се запълва до минимума като границата на реалните данни се определя то тяхната записана дължина.
Identification – 16-битово поле, съдържащо номера на потока, към който принадлежи пакета. В случаите, когато входните данни от по-високо ниво (TCP, UDP) са с по-голям размер от необходимия за IP пакет, те се фрагментират и на получените пакети се назначава един и същи идентификатор на потока, за да може при дефрагментацията да се определи тяхната принадлежност.
Flags – 3-битово поле, указващо дали може да бъде фрагментирана единицата.
бит 0 – запазен. Стойността е 0.
бит 1 – (DF), 0=позволява фрагментация, 1=не позволява фрагментация.
бит 2 – (MF), 0=последен фрагмент, 1=не е последен фрагмент.
Пристигащите пакети, които са част от една фрагментация (общо ID) се буферират в приемника до пълното им приемане, след което се дефрагментират.
Fragment Offset – 13-битово стойност, позволяваща да се определи позицията на пакета в потока. Стойността на първия пакет е 0, а на останалите се изчислява според дължината на MTU(Maximum Transmission Unit). На тази стойност се базира подредбата на пакетите.

Time to Live – 8-битово поле, чиято стойност се намалява при всяко преминаване през маршрутизатор. Целта е пакетът със стойност TTL=0 да се премахне от мрежата като недоставен. Това действие се извършва от маршрутизатора. Той не може да препраща пакети с TTL<=1. Всяка единици от TTL се разглежда като секунда, която на практика е напълно достатъчна за обработка на пакета от маршрутизатора. Междинното устройство, при което TTL=0 изпраща ICMP съобщение до подателя за своето действие.
Protocol – 8-битово поле, заемащо стойности 06(TCP) и 17(UDP), което указва протокола от по-горен слой, за който е предназначен пакета. Други кодове: (89) OSFP, (2) IGMP, (1) ICMP.
Header Checksum – 16-битово поле, което се използва за проверка на целостта на хедъра на предавания пакет. Преизчислява се във всяко междинно устройство, понеже някои от полетата на IP хедъра се променят по време на пътуването (например, TTL). Грешната стойност е причина за премахване на пакета. IP не уведомява изпращача за това си действие, а разчита липсата на пакета да се открие от по-горни слоеве.
Source Address – 32-битово поле, което съдържа адреса на източника.
Destination Address - 32-битово поле, което съдържа адреса на получателя.
Options – поле с променлива дължина, дефиниращо незадължителни опции.
Padding – поле с променлива дължина. Служи за допълване до 32 бита.

На фигура 142 са представени стойностите на заглавната част на IP пакет прехванати чрез мрежовия анализатор Wireshark.



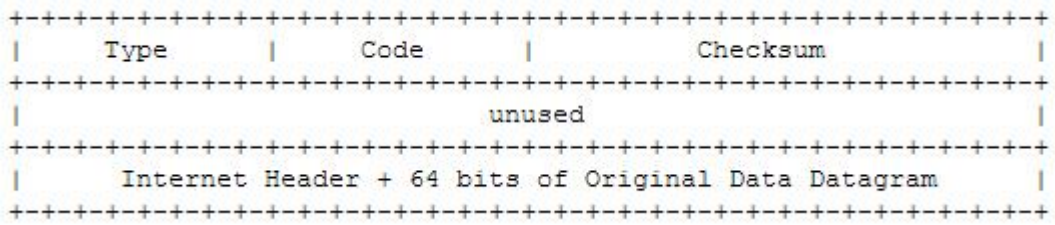
фигура 142 *Заглавна част на IP пакет (Wireshark)*

Анализът показва, че пакетът принадлежи на протокол, версия IPv4. Дължината на заглавната част е 20 байта, а тоталната – 40 байта. Номерът на потока, указващ принадлежността на пакета е 12046, а отместването спрямо началото е 0 т.е това е първи и единствен (More_fragments=0) нефрагментиран (Don't_fragment=1) пакет. Данните, които пренася са от протокол TCP (06) като адресът на източника е 62.233.57.243, а адресът на доставка – 169.168.88.15.

9.5.2. ICMP (Internet Control Message Protocol)

ICMP се счита за неделима част от IP, защото използва неговите услуги. Чрез него крайните хостове и маршрутизатори обменят служебна информация и съобщения за грешки. Полетата на този протокол следват хедъра на IPv4 и съдържат 8 байта заглавна част и променлива дължина на данните. Първите 4 байта имат фиксирана структура, а останалите 4 байта зависят от типа на пакета. В IP хедъра типът на протокола има стойност 1. ICMP (описан в RFC 792) поддържа 15 типа съобщения. От тях типове 3,4,5,11 и 12 са съобщения за грешки, а останалите са заявки, които изискват информация. Например, командата ping, предназначена за тестване на физическата свързаност между два хоста, използва съобщения от тип 8 (Echo Request) и тип 0 (Echo Reply), за да генерира ICMP запитване към целевия хост, който от своя страна трябва да отговори с ехо-отговор. Част от дефинираните типове са:

- Тип 3 (Destination Unreachable Message) – местоназначението не може да бъде достигнато. Разпределението на полетата е показано на фигура 143.



фигура 143 *Полета за Тип 3*

Type – 3;
Code – описва типа на възникналия проблем. Част от заеманите стойности са:
0=net unreachable – мрежата не може да бъде достигната;
1=host unreachable – хостът не може да бъде достигнат;
2=protocol unreachable – протоколът не може да бъде достигнат;
3=port unreachable – портът не може да бъде достигнат;

4=fragmentation needed and DF set – необходима е фрагментация, но е установен битът за забрана на фрагментацията;
5=source route failed – неуспешен път;
Checksum – служи за проверка на ICMP хедъра. Начинът на използване е описан в RFC 1071.

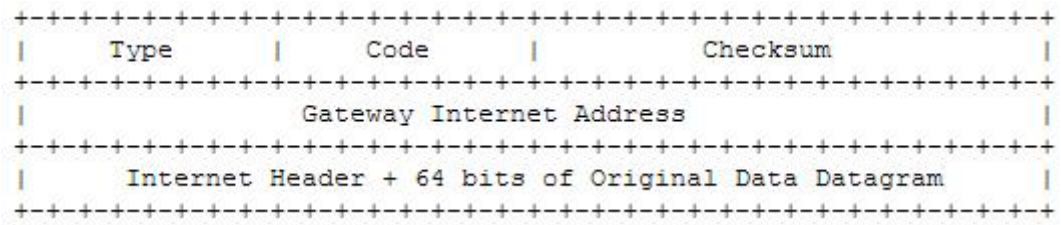
Съобщения с Code=0,1,4 и 5 се издават от маршрутизаторите, а със стойности 2 и 3 от крайните хостове.

- Тип 4 (Source Quench Message) – съобщението може да се генерира от маршрутизатор при недостиг на буферно пространство за сформиране на опашка от пакети за препредаване или от хоста-получател, ако не може да поддържа скоростта за обработка, поради недостиг на ресурси. Разпределението на полетата съвпада с фигура 143.

Type – 4;
Code – приема стойност 0.

Съобщения с Code=0 се издават от маршрутизаторите и крайните хостове.

- Тип 5 (Redirect Message) – информира изпращащия хост за правилния адрес на шлюз, при грешно насочен пакет. Разпределението на полетата е показан на фигура 144.

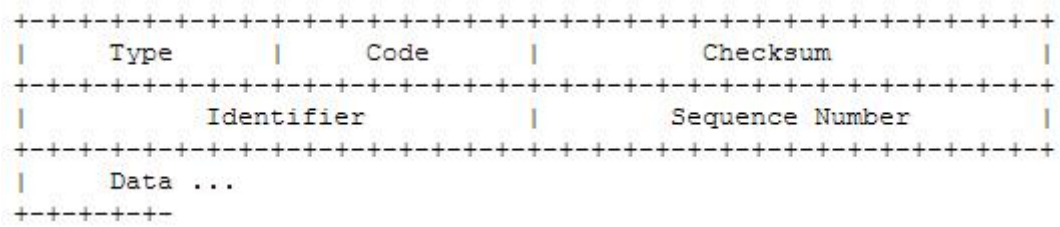


фигура 144 *Полета за Тип 5*

Type – 5
Code – описва типа на грешката от пренасочване. Стойностите са:
0=пренасочване на пакет за мрежа;
1= пренасочване на пакет за хост;
2= пренасочване на пакет за ToS и мрежа;
3= пренасочване на пакет за ToS и хост;
Gateway Internet Address – адрес на правилния шлюз.

Съобщението се генерира само от маршрутизатор към хоста-подател.

- Тип 8 (Echo Message) и Тип 0 (Echo Reply Message) – двете съобщения се използват заедно за тестване на свързаност между хостове. Разпределението на полетата е показан на фигура 145.



фигура 145 *Полета за Тип 0 и 8*

Type – 0 или 8;
Code – приема стойност 0;
Identifier– помощен идентификатор, служещ за установяване на съответствие между ехо-запитване и ехо-отговор;
Sequence Number – стойност, която може да се увеличава при всяко следващо ехо-запитване.

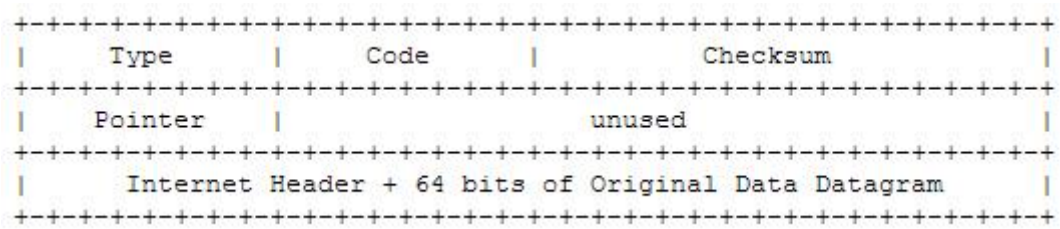
Съобщения с Code=0 се издават от маршрутизаторите и крайните хостове.

- Тип 11 (Time Exceeded Message) – съобщение за превишено време. Разпределението на полетата съвпада с фигура 143.

Type – 11
Code – стойностите са:
0=изтекло време по време на предаване
1=изтекло време за дефрагментация на пакета

Съобщения с Code=0 се издават от междинните устройства, а със стойности 1 от крайните хостове.

- Тип 12 (Parameter Problem Message) – даден параметър не може да бъде интерпретиран. Разпределението на полетата е показан на фигура 146.



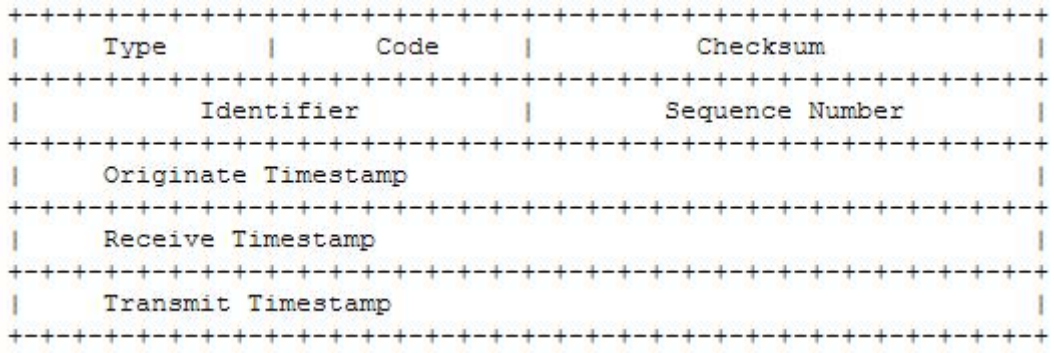
фигура 146 *Полета за Тип 12*

Type – 12
Code – приема стойност 0 при грешка.

Pointer – оказва байта от хедъра, където е възникнала грешката. Например, 1 показва проблеми в полето Type of Service.

Съобщения с Code=0 се издават от междинните устройства и крайните хостове.

- Тип 13 (Timestamp Message) и Тип 14 (Timestamp Reply Message) – двете съобщения се използват заедно за времева синхронизация. Разпределението на полетата е показан на фигура 147.

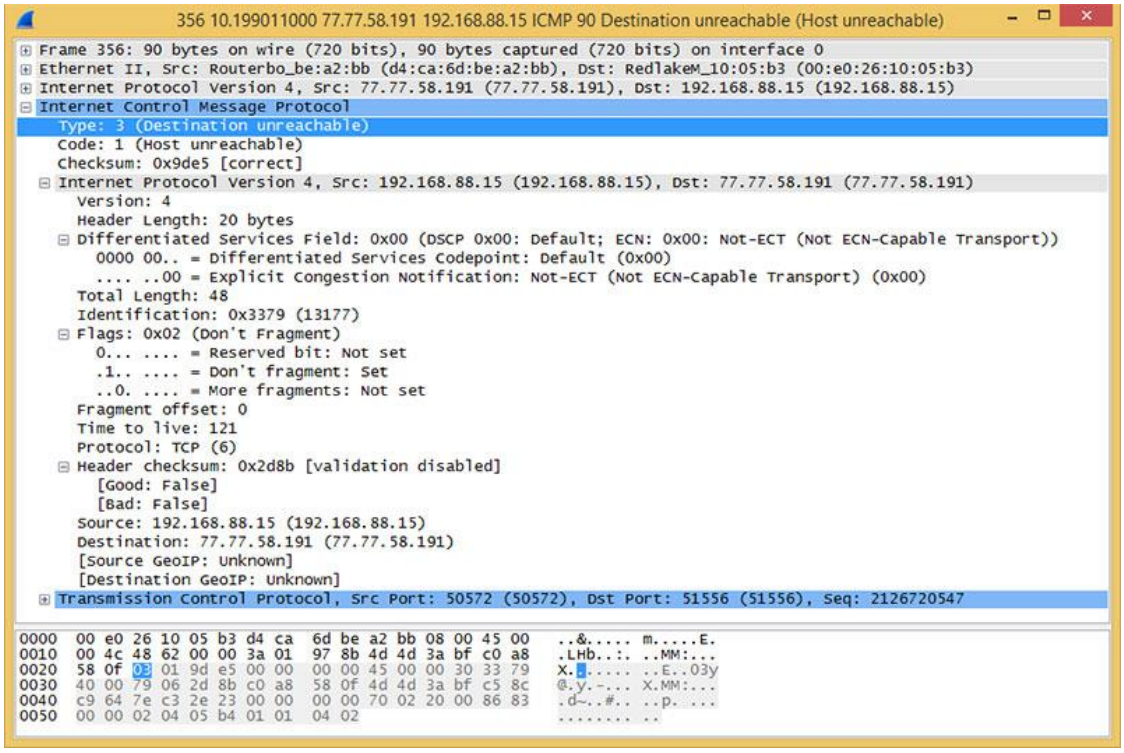


фигура 147 *Полета за Тип 13 и 14*

Type – 13 или 14;
Code – приема стойност 0;
Identifier– помощен идентификатор, служещ за установяване на съответствие между заявка и отговор;
Sequence Number – стойност, която може да се увеличава при всяка следваща заявка;
Originate Timestamp – първоначален час на изпращане;
Receive Timestamp – час на приемане;
Transmit Timestamp – час на предаване

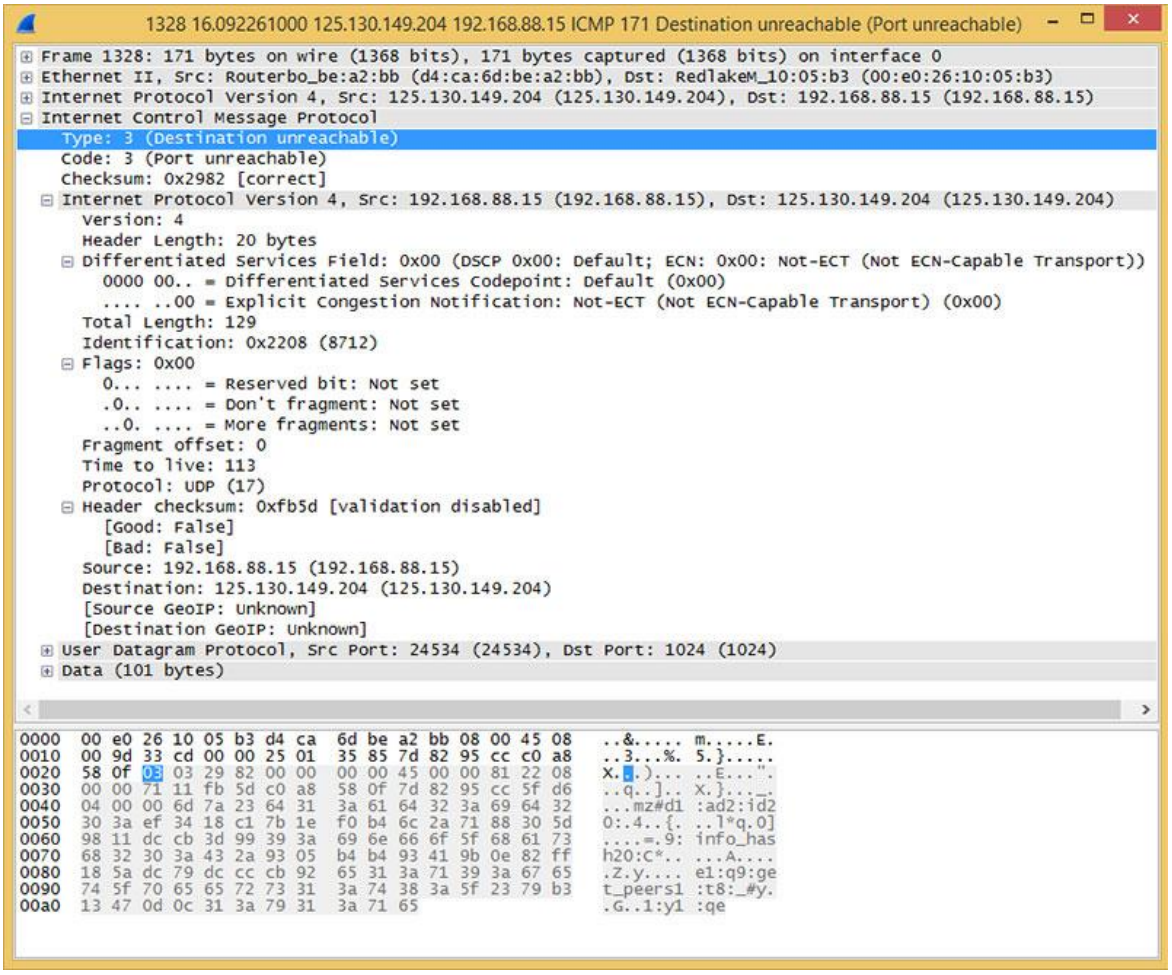
Съобщения с Code=0 се издават от маршрутизаторите и крайните хостове.

Примерни стойности за заглавните части на ICMP хедър са представени на фигура 148. От нея се вижда, че типа на съобщението е 3 (Destination Unreachable Message), а кодът е 1 (Host unreachable). Полето Checksum за проверка на ICMP хедъра показва, че записаната в него стойност е коректна. Следва IP и TCP заглавната част на изпратения пакет, предизвикал ICMP съобщението.



фигура 148 *ICMP заглавна част*

Следващият пример е за съобщение тип 3 (Destination Unreachable Message) и код 3 (Port unreachable). Този ICMP пакет е предизвикан от опита за пренос на UDP (17) транспортна единица към хост с адрес 125.130.149.204.



фигура 149 *ICMP заглавна част*

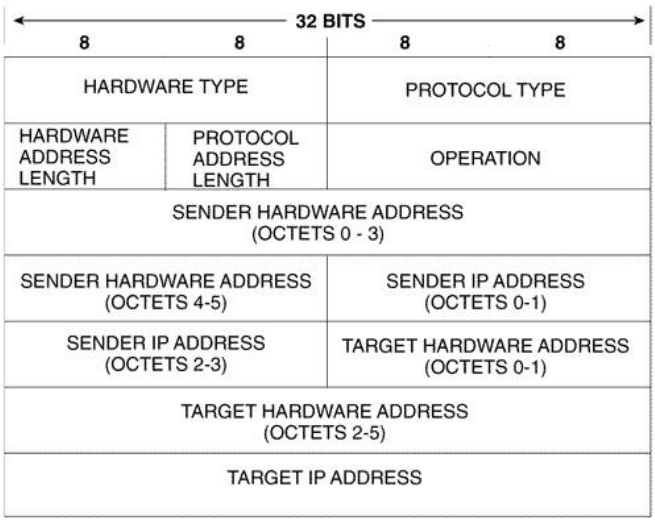
9.5.3. ARP (Address Resolution Protocol)

ARP е протокол за преобразуване на адреси. Превръща логическите адреси (например, 32-битови IP адреси) от мрежовия слой във физически адреси от канален слой (например, MAC адреси) [RFC 826]. За целта всеки хост поддържа ARP кеш таблица, където съхранява съответствията между IP и MAC адрес, научени динамично по време на комуникацията с други хостове или въведени статично от администратора на системата. ARP използва броадкасти до хостовете в локалния сегмент за определяне на дадено съответствие, което добавя като запис в ARP таблицата на хоста за бъдещо използване. Валидността на записите може да се контролира чрез няколко механизма:

- таймаут – при добавяне на запис в таблицата се определя време на валидност, след което той се премахва;

- периодични уникаст запитвания – изпращат се периодични уникаст запитвания към регистрираните хостове. Ако отдалеченият хост не отговори, записът се премахва от кеша;
- уведомяване от протокол – ако протокол от по-горен слой установи проблеми при доставката, той уведомява активния ARP процес в хоста, който от своя страна премахва записа за отдалечения хост от таблицата му.

Структурата на заглавната част на ARP и описанието ѝ е показано на фигура 150.

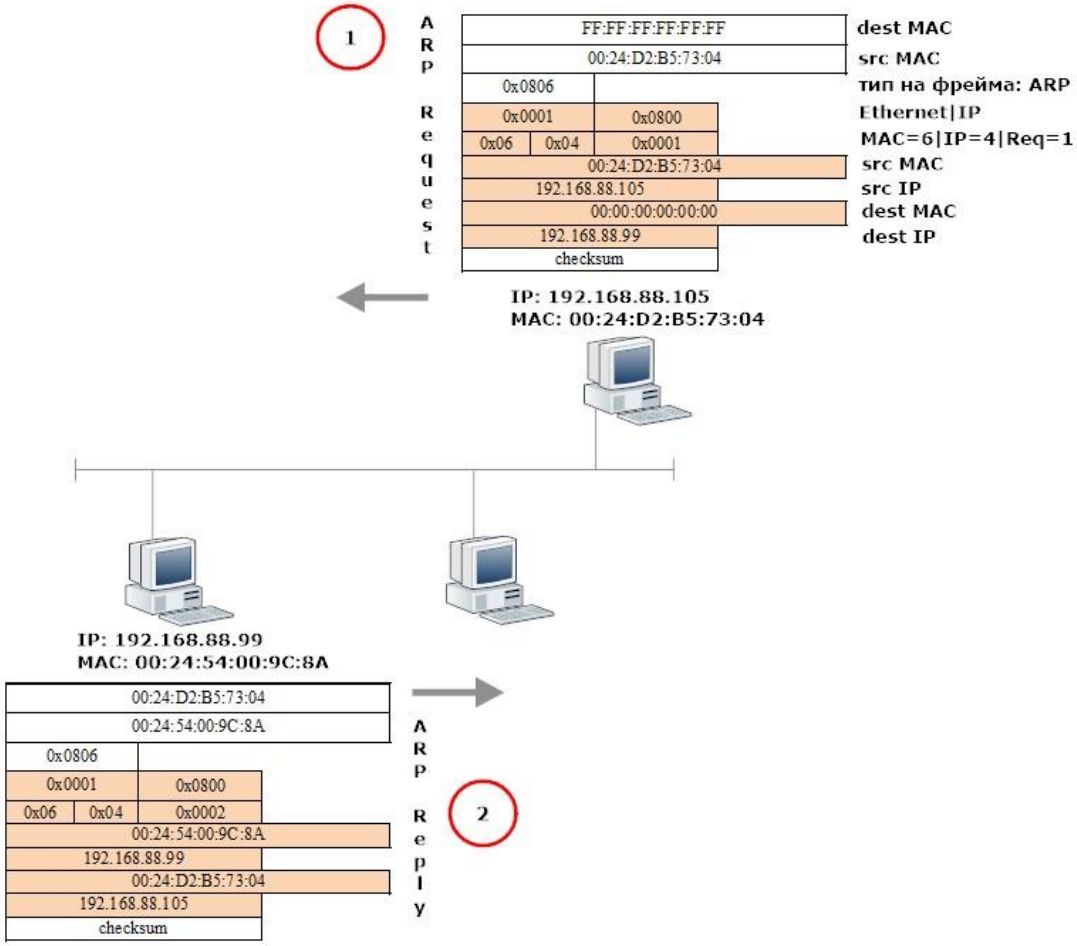


фигура 150 *Заглавна част на ARP*

Hardware type – 2-байтово поле, идентифициращо типа на хардуера (например, Ethernet, Token-Ring или друг тип мрежа). За Ethernet това поле има стойност 0x0001;
Protocol type – 2-байтово поле, идентифициращо типа на протокола от мрежовия слой (например, IP=0x0800, IPX=0x8137);
Hardware length – 1-байтово поле, задаващо дължината в байтове на хардуерния адрес (например, Ethernet има 6-байтов MAC адрес);
Protocol length - 1-байтово поле, задаващо дължината в байтове на протоколния адрес от мрежово ниво, който се преобразува. Например, за IP=0x0800 полето трябва да съдържа стойност 4;
Operation - 2-байтово поле, идентифициращо типа на извършваната операция:
0x0001=ARP request;
0x0002=ARP reply;
0x0003=RARP request;
0x0004=RARP reply.
За разграничаване на типа на кадъра (ARP или RARP) в полето Ethertype на DLC хедъра (фигура 151) се записва 0x0806 (ARP) или 0x8035 (RARP).
Sender hardware address – 6-байтово поле за хардуерния адрес на изпращача;
Sender protocol address – 4-байтово поле за логическия адрес на изпращача;
Target hardware address - 6-байтово поле за хардуерния адрес на получателя;

Target protocol address - 4-байтово поле за логическия адрес на получателя.

фигура 151 *DLC ARP кадър*

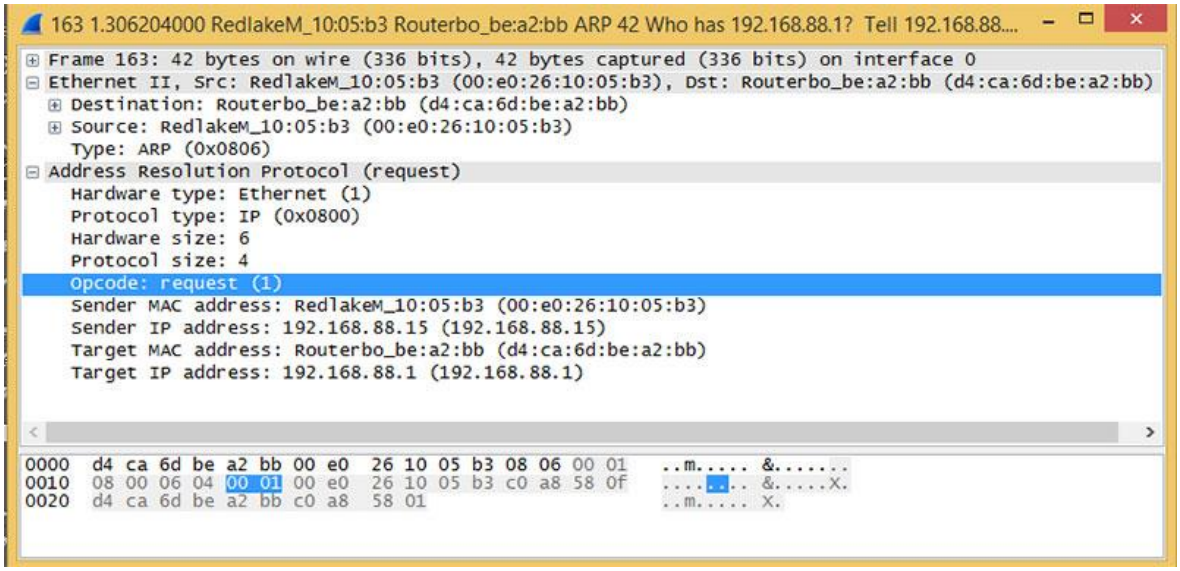


фигура 152 *ARP заявка и отговор*

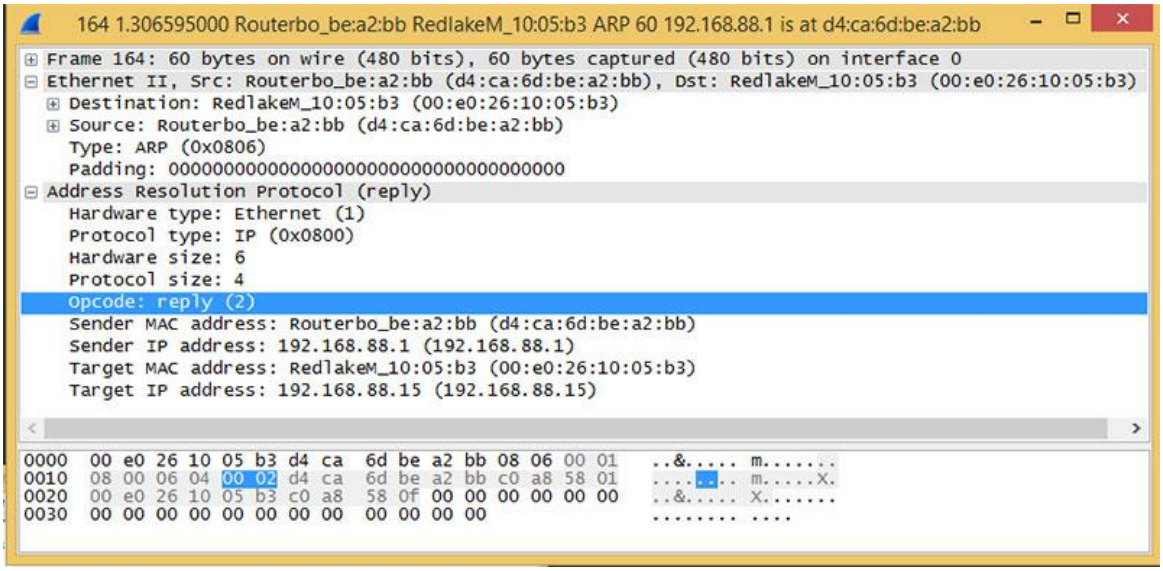
На фигура 152 е представена ARP заявка (Operation=0x0001) и ARP отговор (Operation=0x0002). Компютър с IP адрес 192.168.88.105 изпраща в локалния сегмент ARP броадкаст (destMAC=FF:FF:FF:FF:FF:FF) за установяване на непознат MAC адрес, съответстващ на локален IP адрес 192.168.88.99. Търсеният хост отговаря на запитването чрез ARP отговор, като предоставя физическия си адрес на хоста-изпращач. Втората възможност е ако изпращащият хост е установил, че получателят не се намира в същия сегмент. Тогава неговият ARP броадкаст ще бъде

предназначен за установяване на физическия адрес на локалния шлюз (ако не го знае все още), който да препрати пакета до хоста-местоназначение.

Пример за периодично уникаст запитване е представен на фигура 153 и фигура 154. Типа на полето Opcode на фигура 153 е със стойност 0x0001 (ARP request), а за фигура 154 е 0x0002 (ARP reply). За разграничаване типа на кадъра (ARP или RARP) в секцията Ethertype II полето Type заема стойност 0x0806 (ARP). Преобразуването се извършва между адреси от хардуерен тип Ethernet (Hardware type: 0x0001) с дължина 6 байта (Hardware size: 6) и протокол IP (Protocol type: 0x0800) от мрежовия слой с дължина на адреса 4 байта (Protocol size: 4).

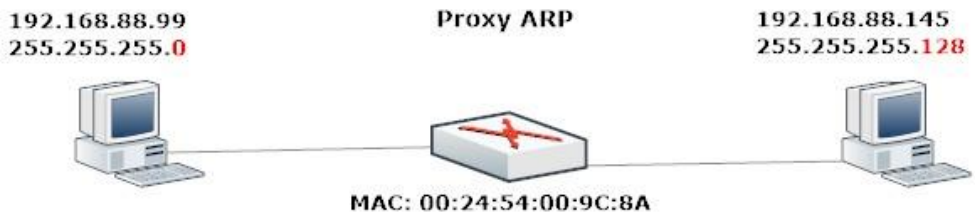


фигура 153 *ARP заявка*



фигура 154 *ARP отговор*

В някои случаи е възможно използването на Proxy ARP, което позволява на друго устройство (например, маршрутизатор) да отговаря на ARP запитвания от името на отдалечен хост, намиращ се в друга подмрежа. Това осигурява възможност за прозрачна комуникация между отдалечени хостове. Например, ако даден хост-изпращач е конфигуриран неправилно и се обърка при изпращането на ARP запитване към хост извън локалния сегмент то маршрутизаторът, конфигуриран като Proxy ARP, ще отговори със собствения си физически адрес. Това позволява пакетът да бъде препратен след това към правилния хост-местоназначение.

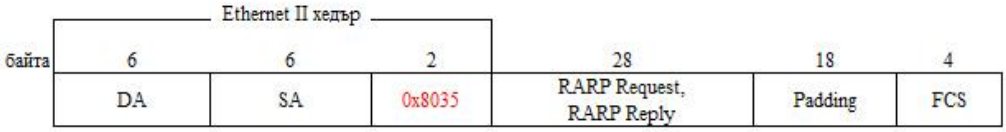


фигура 155 *Proxy ARP*

Подобен пример е представен на фигура 155, където хост 192.168.88.99 от мрежа 192.168.88.0 иска да комуникира с хост 192.168.88.145 от мрежа 192.168.88.128. Мрежовата маска на първия хост (255.255.255.0) е сгрешена. Това го обърква при пресмятането на принадлежността на адрес 192.168.88.145. За него този адрес е част от локалния сегмент и затова изпраща локален ARP броадкаст за установяване на физическия му адрес. Това запитване не стига да хоста-местоназначение и той не отговаря. Маршрутизаторът, конфигуриран като Proxy ARP, приема това запитване и отговаря от името на целевия хост като изпраща собствения си хардуерен адрес. Изпратеният към него пакет бива препратен към хоста-местоназначение.

9.5.4. RARP (Reverse Address Resolution Protocol)

RARP е протокол за динамично преобразуване на физическите адреси на хостове в логически адреси от мрежово ниво (например, IP адреси). Той извършва противоположно действие на ARP протокола. Използва същия формат на хедъра, както ARP протокола. Кадърът, пренасящ RARP, се разпознава по типа на полето Ethertype=0x8035 (фигура 156).



фигура 156 *Ethertype=0x8035*

Протоколът е предназначен за назначаване на логически адреси на бездискови станции. За целта се използват RARP сървъри, където в статични таблици се съхраняват съответствията между хардуерни и логически адреси. За поддържането им се грижат мрежови администратори. Подобно на ARP и този протокол използва броадкасти. Това означава, че във всеки локален сегмент трябва да има поне по един RARP сървър, който да отговаря на такъв тип запитвания. Недостатъците на RARP са свързани с необходимостта от:

- такъв тип сървъри за всеки един сегмент, поради невъзможността маршрутизаторите да препращат RARP броадкасти;
- администратор за създаване и поддържане на статичните асоцииращи таблици.

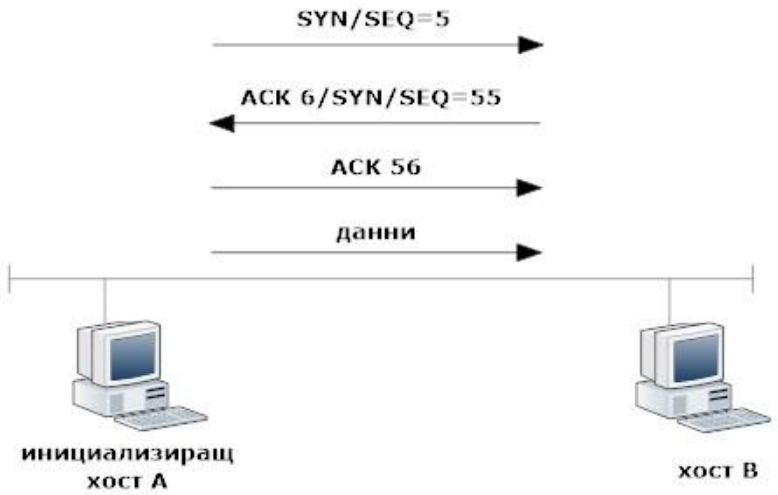
RARP не е единственият протокол за преобразуване на хардуерни в логически адреси. Представители на този тип асоцииране са още BOOTP и DHCP, предназначени да заменят RARP. Всеки един от тези протоколи се стреми да отстрани проблемите, съществуващи при предшественика му. В момента актуалният протокол е DHCP.

9.5.5. TCP (Transfer Control Protocol)

TCP е протокол, осигуряващ надеждно предаване на сегменти между съответните процеси на двата комуникиращи хоста. Той назначава пореден номер за всеки предаден байт и очаква потвърждение от приемащия хост за получените байтове. Това забавя доставката на данни за сметка на тяхната сигурност, която се характеризира със следните действия:

1. Създаване на логическо съединение (сесия) между двата комуникационни процеса (Session setup) – този тип съединение се изгражда между портовете на двата комуникиращи хоста. Използването на портове позволява на TCP да поддържа множество логически съединения, едновременно, между два хоста или между хост и множество хостове. Този процес се нарича мултиплексиране.

При изграждането на сесия се използва подход, наречен трикратно ръкостискане (three-way handshake). При този механизъм, целевият хост получава SYN сегмент от хоста-източник, на който отговаря с ACK/SYN сегмент и очаква неговото ACK потвърждение от инициализиращия хост. SYN сегментите дават възможност на комуникиращите хостове да установят стартови параметри за комуникацията – размер на прозореца (буфер), първоначален пореден номер (специфичен за всеки хост), максимален размер на сегмент. На фигура 157 е илюстрирано трикратното ръкостискане.



фигура 157 *Трикратно ръкостискане*

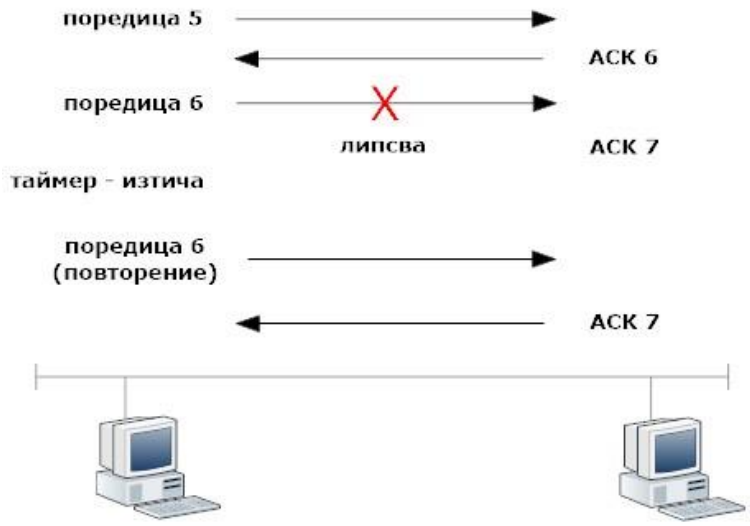
- Хост A изпраща SYN сегмент с включен начален пореден номер SEQ=5 (Sequence Number);
- Хост B потвърждава с ACK (5+1=6), включен бит SYN и началния си пореден номер SEQ=55;
- Хост A потвърждава ACK/SYN сегмента с ACK (55+1=56);
- Извършва се трансфер на данни след установяване на връзката. TCP използва пореден номер за всеки байт прехвърлени данни, които трябва да бъдат потвърдени от приемащия хост. Полето SEQ идентифицира първият байт от сегмента, а полето ACK съдържа очаквания следващ пореден номер, като потвърждава всички данни до него. Например, ако Хост A предаде 3 сегмента с по 15 байта данни и началният пореден номер е 30, то следващия очакван пореден номер, заложен в ACK пакета, ще бъде 30+3.15=75;

2. Контрол на последователността на данните в сегмента (Sequencing);

3. Контрол на потока от данни (Flow control) – гарантира, че входящият трафик няма да запълни буферите на приемащия хост и той ще може да обработи потока от данни, както и да отговори на запитвания от страна на предаващия хост. Механизмът, който се използва е познатият плъзгащ се прозорец (sliding window). Всеки хост поддържа такъв прозорец и контролира размера му спрямо моментните си възможности. Чрез него приемащият хост указва количеството данни, което може да буферира. Максималната стойност е 65535 байта и се определя от размера на полето, което е 16-битово (Window). Размерът на прозореца, потвържденията и поредните номера са байтово базирани, а не сегментно;

4. Поддържане на връзката при липса на данни за предаване – използва се служебно съобщение (keepalive) за поддържане на връзката. То не съдържа данни от по-горен слой, което означава, че полето за дължина е 0 и следващият ACK номер не се увеличава. Ако такива връзки не се затворят се увеличава натоварването на мрежата, особено в случаите, когато са много на брой;

5. Потвърждение на правилно приетите данни (Acknowledgement)– потвърждението е механизъм, позволяващ на хостовете да определят кога има загуба на данни. Приемащият хост не изпраща ACK потвърждение за изгубен пакет. При неполучаване на потвърждение за определен период от време (използва се таймер), изпращащият хост повтаря предаването на непотвърдените данни, които се съхраняват в т.нар. TCB буфер (блок за контрол на предаването) (фигура 158);



фигура 158 *Препредаване на липсваща поредица*

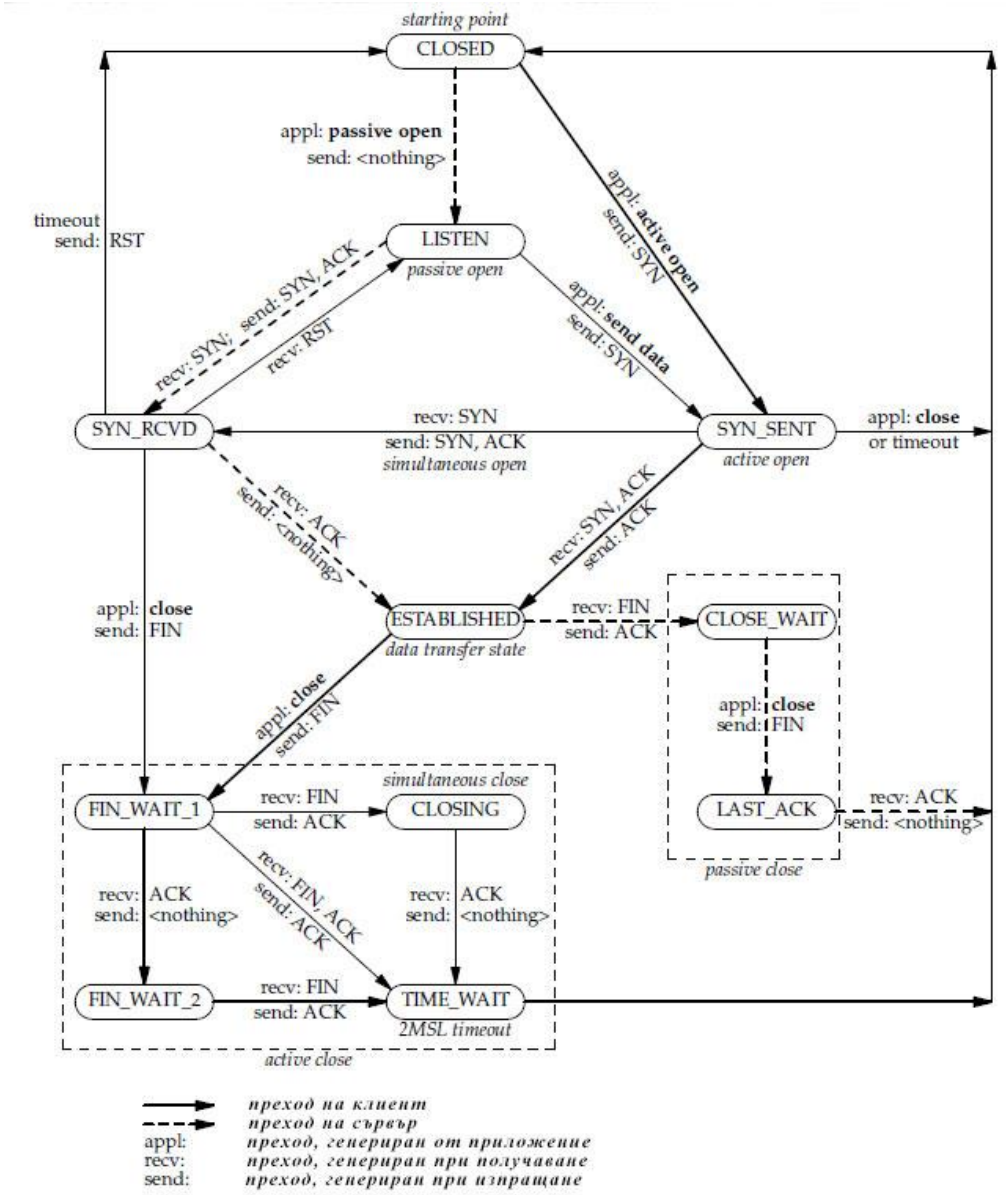
6. Разпадане на логическото съединение (Session teardown) – процесът е подобен на този за установяване на сесия. Използва се трикратно ръкостискане, където SYN сегментът е заменен с FIN. Затварящият хост изпраща FIN сегмент към другата страна за затваряне на сесията. Приемащият хост потвърждава с ACK/FIN отговор, които включва и собствен FIN, очакващ потвърждението му от хоста-инициатор. След тази процедура сесията се затваря. Състоянията, през които преминават двете страни, използващи TCP съединението са:

- за клиента – CLOSED, SYN-SENT, ESTABLISHED, FIN-WAIT-1, CLOSE-WAIT, FIN-WAIT-2, CLOSING, LAST-ACK, CLOSED;
- за сървъра – CLOSED, LISTEN, SYN-RESEIVED, ESTABLISHED, FIN-WAIT-1, CLOSE-WAIT, FIN-WAIT-2, CLOSING, TIME-WAIT, CLOSED.

означение	**състояние**
LISTEN	състояние на сървъра, очакващ създаване на съединение
SYN-SENT	състояние на клиента, изчакващ отговор на SYN заявката си за създаване на съединение
SYN-RECEIVED	състояние на сървъра, очакващ потвърждение ASK на изпратения от него SYN
ESTABLISHED	състояние и на двете страни при изградено и използващо се съединение
При закриване на съединение инициатори могат да бъдат и двете страни	
FIN-WAIT-1	състояние на инициатора за закриване на съединението след изпращане на заявката FIN. Очаква потвърждение от другата страна
CLOSE-WAIT	състояние на отсрещната страна след отговор ACK на заявката за закриване (FIN)
FIN-WAIT-2	състояние на инициатора след получаване на ACK потвърждението от другата страна на неговата FIN заявка
LAST-ACK	състояние на отсрещната страна след изпращане на FIN отговор
CLOSING	състояние на двете страни при едновременно затваряне на използваното съединение
TIME-WAIT	състояние на изчакване за определено време преди закриване на съединението

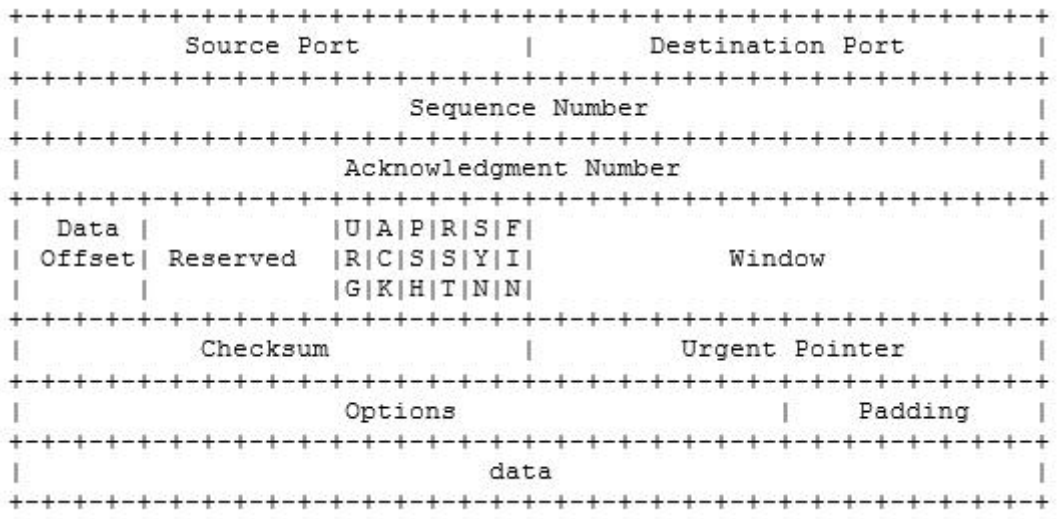
таблица 27 *Състояние на двете страни на TCP съединение*

Състоянията и събитията предизвикващи тяхната промяна са представени на фигура 159.



фигура 159 *TCP състояния*

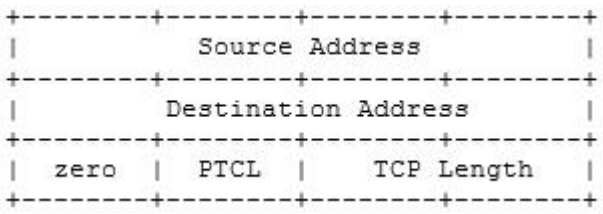
Структурата на TCP сегмента [RFC 793] е показана на фигура 160.



фигура 160 *Структура на TCP сегмента*

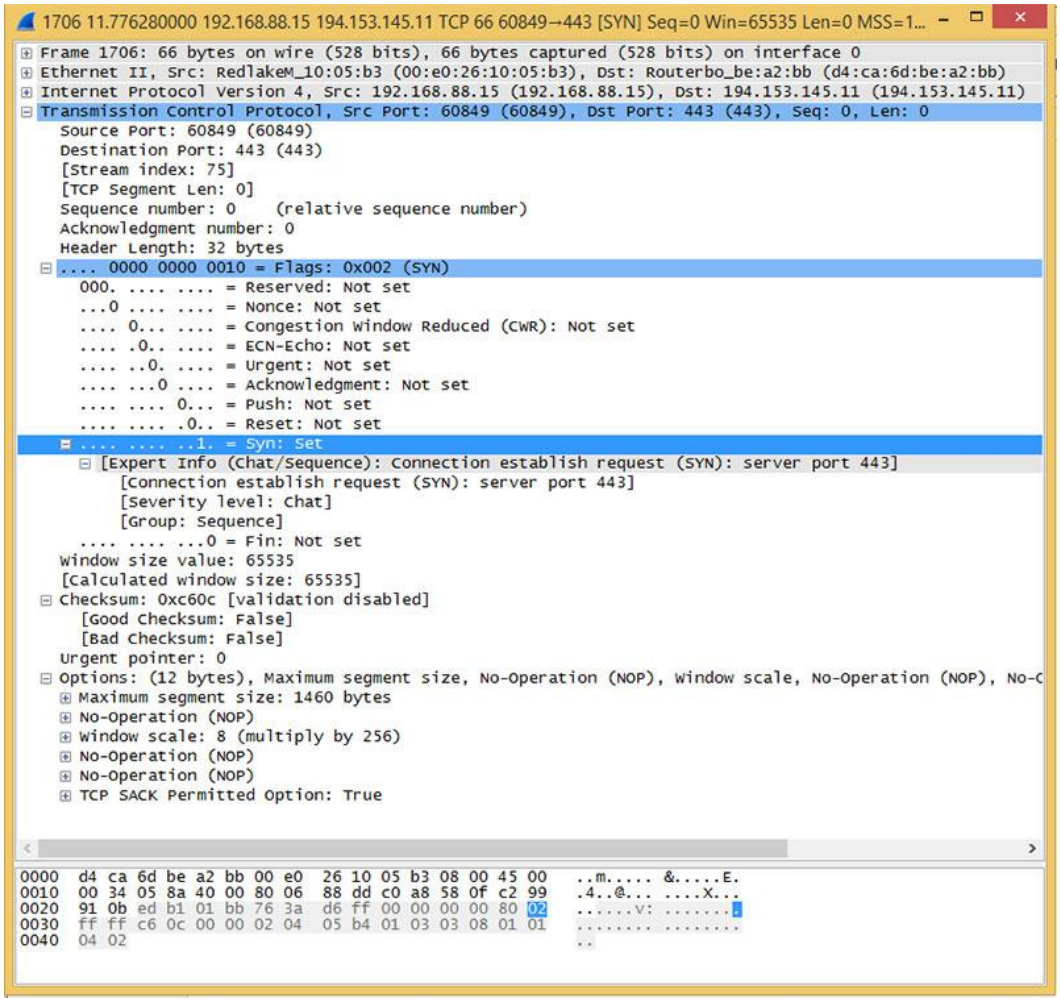
Source Port – 2-байтово поле, идентифициращо изходния порт (сокета) на комуникационния процес на хоста-изпращач;
Destination Port – 2-байтово поле, идентифициращо входния порт (сокета) на комуникационния процес на хоста-приемник;
Sequence Number - 4-байтово поле, съдържащо поредния номер на първия октет от данни в сегмента;
Acknowledgment Number - 4-байтово поле, съдържащо поредния номер на началния октет на следващата последователност от данни;
Data Offset – 4-битово поле, указващо началото на данните, следващи TCP хедъра. Налага се поради променливата дължина на хедъра.
Reserved – 6-битово поле е резервирано и винаги е запълнено с 0.
URG (Urgent) – 1-битово поле, задаващо висок приоритет на данните. Активира Urgent Pointer указателя, сочещ първия байт от сегмента след спешните данни;
ACK (Acknowledgment) - 1-битово поле, задаващо сегмента като потвърждение;
PSH (Push) - 1-битово поле, което при стойност 1 задължава приемащия хост да не задържа пристигащите данни, а да ги изпрати към приложния процес от по-горен слой;
RST (Reset) - 1-битово поле, задаващо стойност 1 ако е необходимо прекъсване на сесията;
SYN (Synchronization) - 1-битово поле, задаващо инициализирането на сесия;
FIN (Finish) - 1-битово поле, задаващо финализирането на сесия от изпращащия хост;
Window - 2-байтово поле, задаващо свободния размер на буфера (в байтове) на приемащия хост. Стойността варира според възможностите на хоста. Ако стойността на прозореца е 0, то този хост не може да приема данни в момента.
Checksum - 2-байтово поле, съдържащо контролната сума изчислена върху TCP хедъра, IP псевдохедъра (фигура 161) и данните;
Urgent Pointer - 2-байтово поле, задаващо байта в сегмента, от където започват не-спешните данни. Включва се при URG=1;
Options – поле с променлива дължина. Включва опции, избрани от изпращащия хост (например, опцията за максимален размер на сегмент - MSS).

Псевдохедърът спомага за откриването на погрешно насочени сегменти. Съхранява се в TCB (transmission control block) буфера и включва показаните на фигура 161 полета от IP хедъра.

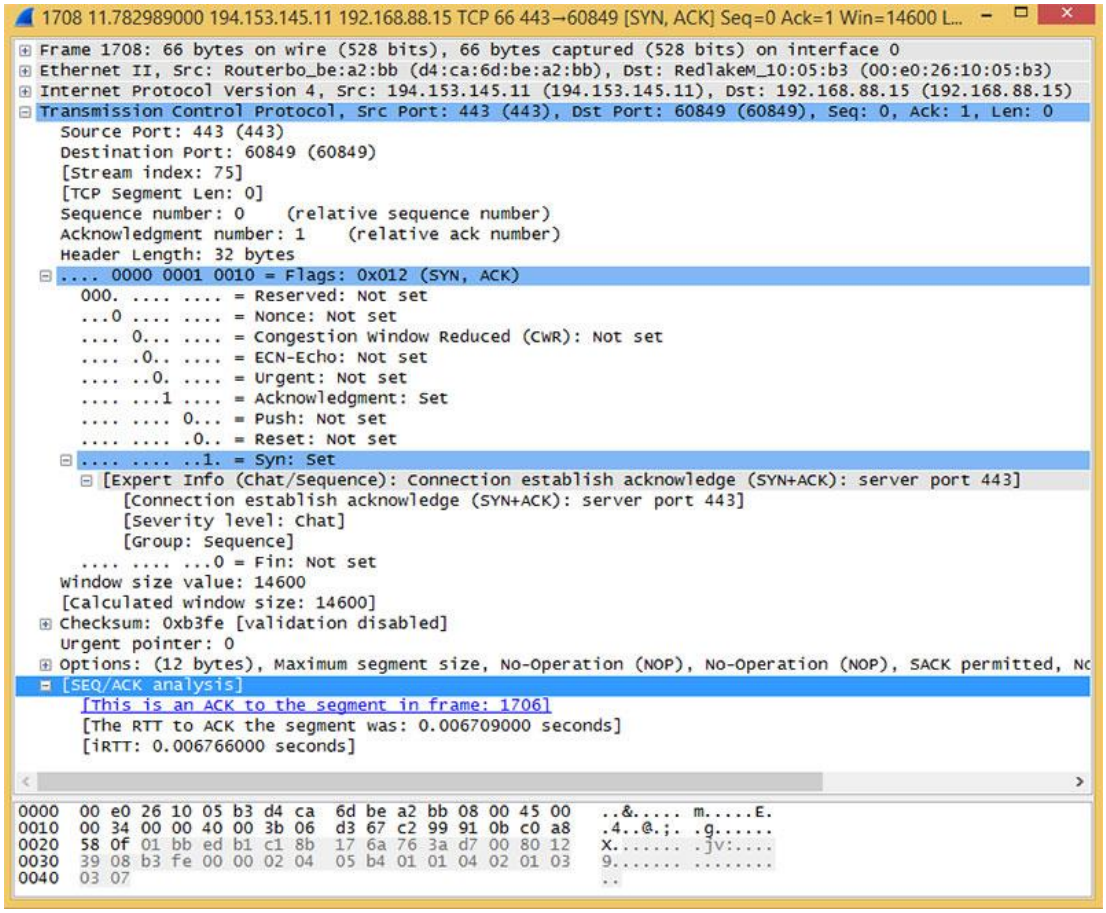


фигура 161 *Псевдохедър*

На следващите фигури са предствени реални TCP сегменти, участващи в една сесия.



фигура 162 *SYN сегмент*

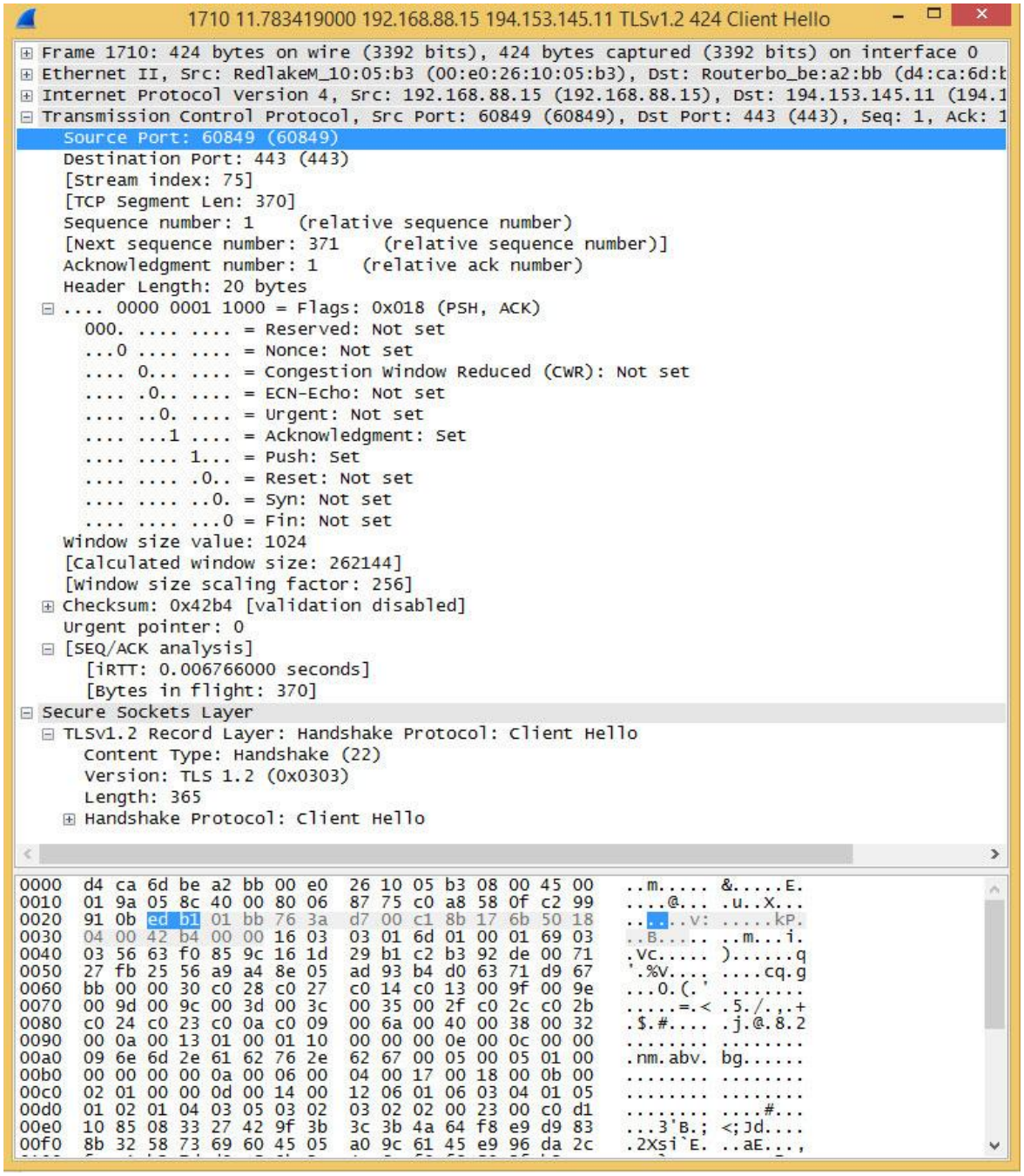


фигура 163 *SYN, ASK сегмент*

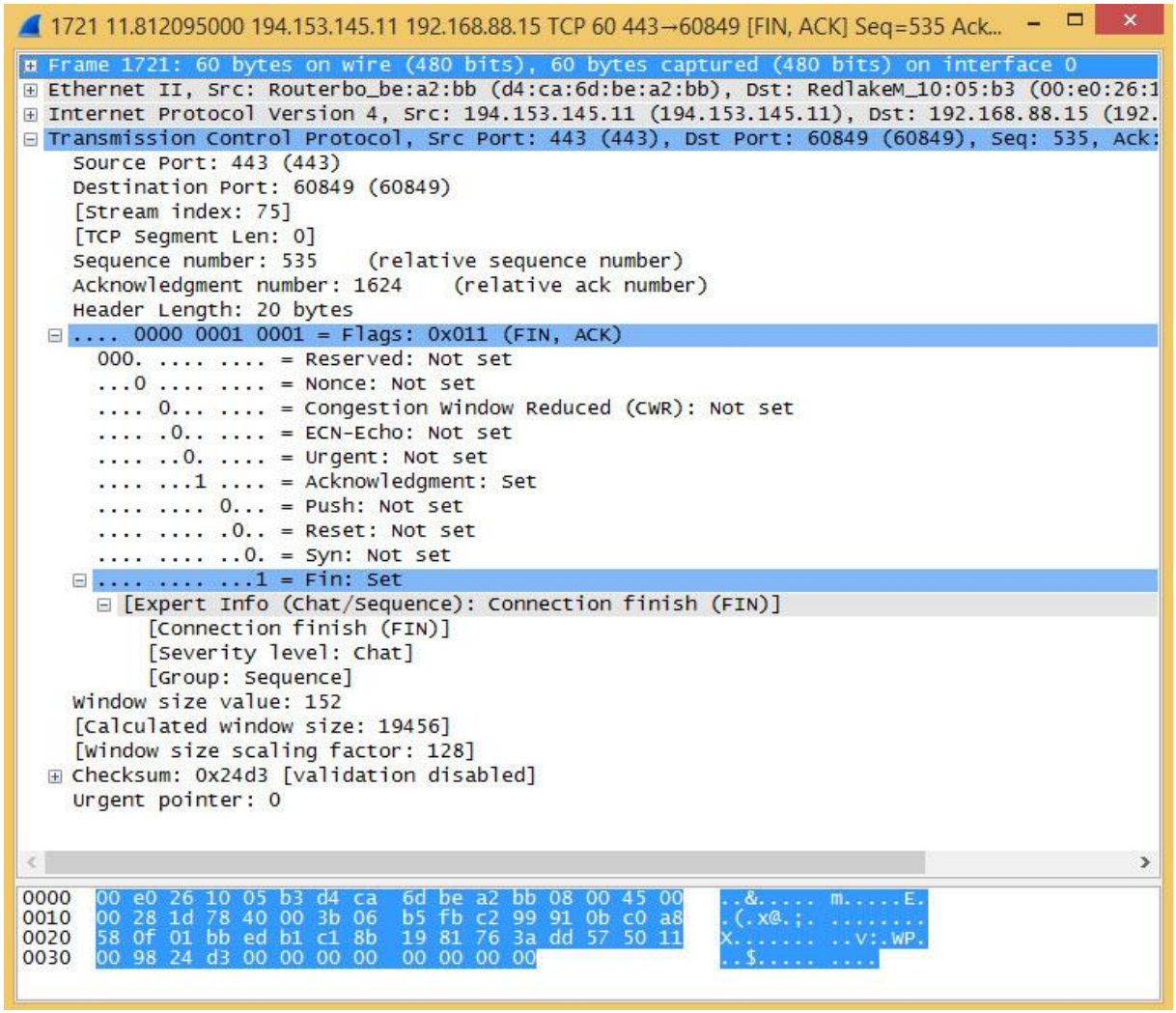
Първият сегмент (фигура 163) е SYN сегмент за инициализиране на TCP сесия. Източникът е хост с адрес 192.168.88.15, а целевият адрес е 194.153.145.11. Вторият сегмент (фигура 163) е [SYN, ASK] сегмент за установяване на сесията. Сегментът от фигура 164 е за трансфер на данни свързани със SSL протокол.

Стойността на SEQ (Sequence Number) е 1, дължината на сегмента е 370 байта, което означава, че следващият SEQ номер трябва да бъде 370+1=371. Изчисляването на стойностите на SEQ и ACK при комуникацията отговаря на следното правило: *SEQ=ACK, ACK=1+предадени байтове на TCP сегмента*.

Останалата последователност от сегменти не е представена. На фигура 165 е показан [FIN, ACK] сегмент за приключване на TCP сесията.



фигура 164 *DATA сегмент*

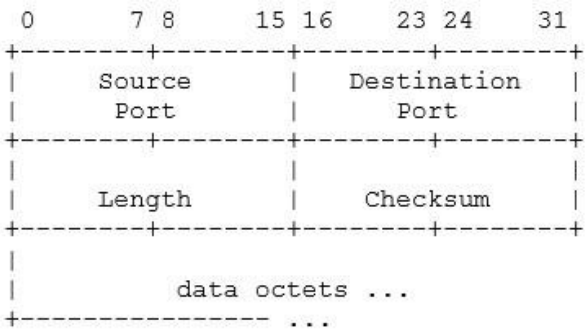


фигура 165 *[FIN, ACK] сегмент*

9.5.6. UDP (User Datagram Protocol)

UDP е протокол, който осигурява бързо, но ненадеждно предаване на сегменти между комуникиращите процеси. В хедъра на IP протокола се идентифицира като протоколен тип 17. За разлика от TCP протокола не създава сесии (не поддържа логически съединения), не потвърждава приетите данни и не контролира техния ред. UDP разчита, че протоколите от по-горен слой ще извършат тези действия.

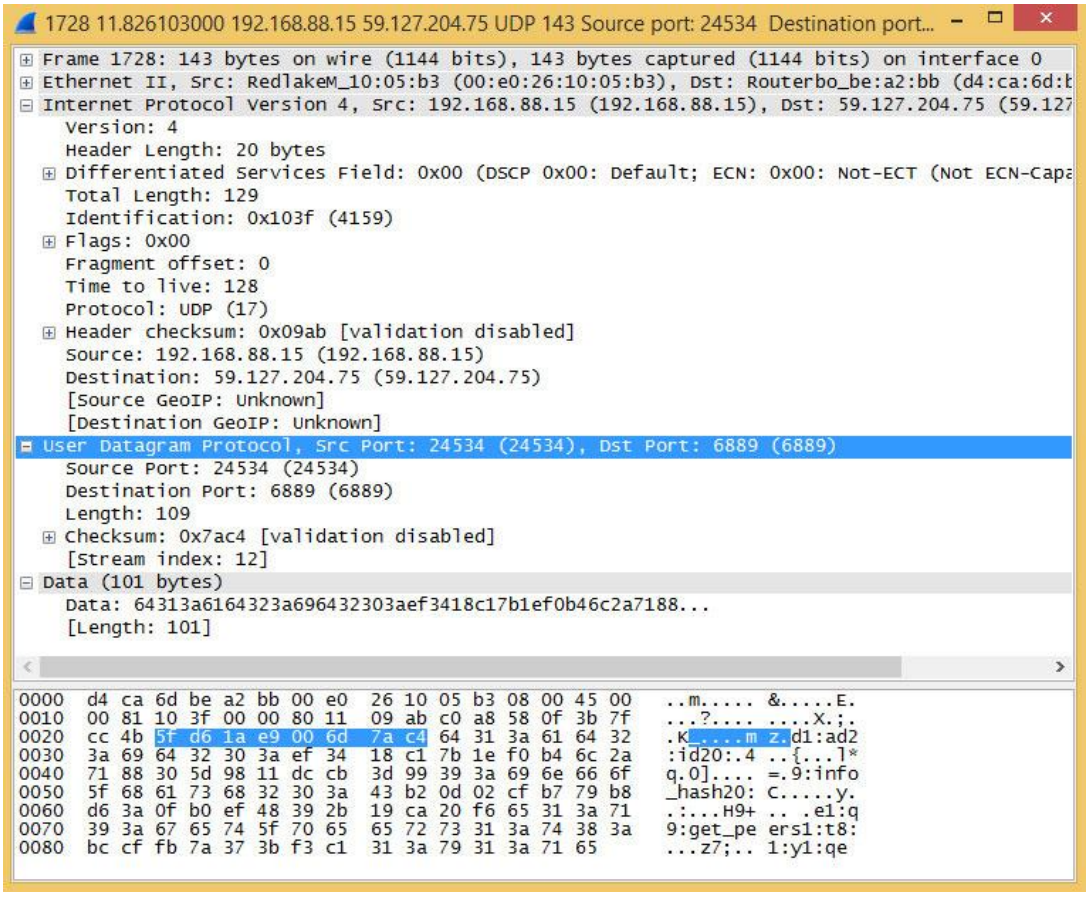
Структурата на UDP сегмента [RFC 768] е показана на фигура 166.



фигура 166 *Структура на UDP сегмента*

Source Port – 2-байтово поле, идентифициращо изходния порт (сокета) на комуникационния процес на хоста-изпращач;
Destination Port – 2-байтово поле, идентифициращо входния порт (сокета) на комуникационния процес на хоста-приемник;
Length – указва дължината на дейтаграмата в байтове и включва хедъра и данните. Минималната дължина е 8 байта.
Checksum – гарантира елементарна проверка за коректно предадена дейтаграма. Методът открива и игнорира повредени единици за данни. Контролната сума проверява за валидност UDP хедъра, данните от по-висок слой и UDP псевдохедъра.

Примерен UDP сегмент е показан на фигура 167.



фигура 167 *Примерен UDP сегмент*

9.5.7. DHCP (Dynamic Host Configuration Protocol)

DHCP е протокол, осигуряващ IP конфигурационни параметри за мрежови устройства. Той се явява подобрена версия на BOOTP. Подобно на BOOTP, маршрутизаторите могат да препращат DHCP заявки и техните отговори. За разлика от RARP и BOOTP, той не поддържа само статични асоцииращи таблици, а използва и пулове от IP адреси, които раздава динамично на хостове и шлюзове. Настройките позволяват да се дефинират пулове от IP адреси, време за лизинг за тези адреси и допълнителни конфигурационни параметри, като адрес на шлюз, DNS сървъри и др. DHCP е базиран на клиент/сървър архитектура. Клиентите изпращат заявка, а DHCP сървърът отговаря с конфигурационни параметри. Методите на раздаване могат да бъдат:

- ръчен – IP адресите са с предварително установено съответствие за конкретно устройство, реализирано от мрежови администратор;
- автоматичен-статичен – IP адресите се задават автоматично, без намесата на администратор, при първоначалното назначаване. Установеното съответствие между IP адресите и хардуерните адреси се запазва постоянно при следващите инициализации;
- автоматичен-динамичен – конфигурационната информация се предоставя за период от време (lease time).

В зависимост от конкретната реализация на протокола, методите могат да се използват отделно или комбинирано.

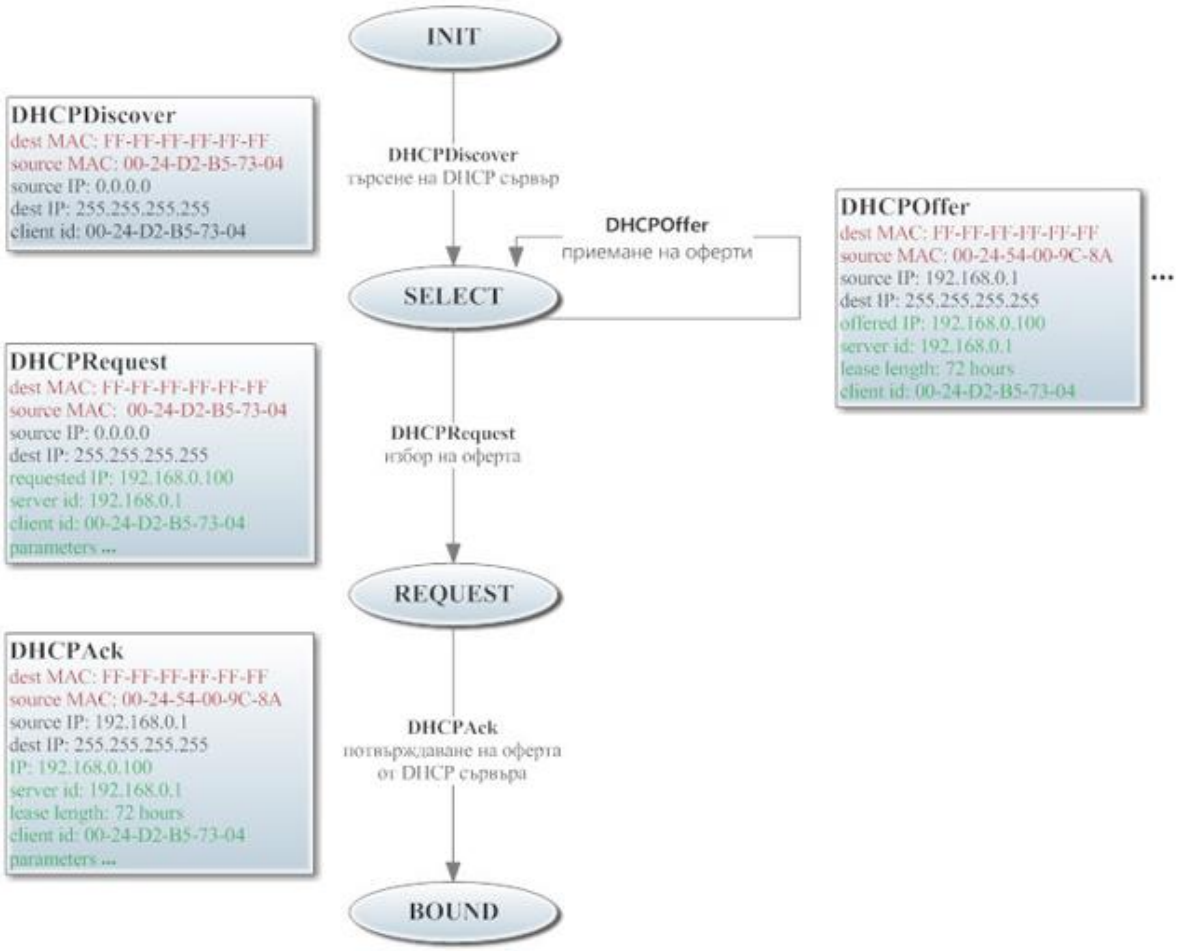
За протокола са дефинирани следните типове съобщения:

1. DHCPDiscover – локален броадкаст за откриване на DHCP сървъри;
2. DHCPOffer – съобщение-отговор от DHCP сървърите, включващо конфигурационни параметри, предлагани от тях. На една заявка от клиент може да отговори повече от един сървър;
3. DHCPRequest – съобщение от клиента, изпращано при потвърждение на конкретно предложение (това означава отхвърляне на всички други такива) или запитване за удължаване на времето на лизинг за използвано IP;
4. DHCPAck – потвърждение от сървъра за предоставянето на конфигурационните параметри;
5. DHCPNak – отрицателно потвърждение от сървъра за неудобрени конфигурационни параметри изискани от клиента;

6. DHCPDecline – отказ, изпращан от клиента за уведомяване на сървъра за вече използвани от друг параметри;
7. DHCPRelease – освобождаване, изпращано от клиента за уведомяване на сървъра, че клиентът не желае моментните зададени IP адрес и конфигурационни параметри (могат да бъдат предоставени на друг клиент);
8. DHCPInform – информиране, изпращано от клиенти, притежаващи ръчно настроени IP адреси, но изискващи допълнителни параметри от DHCP сървъра.

Процесът на договаряне (за IP адрес и конфигурационни параметри) и състоянията на клиента са показани на фигура 168.

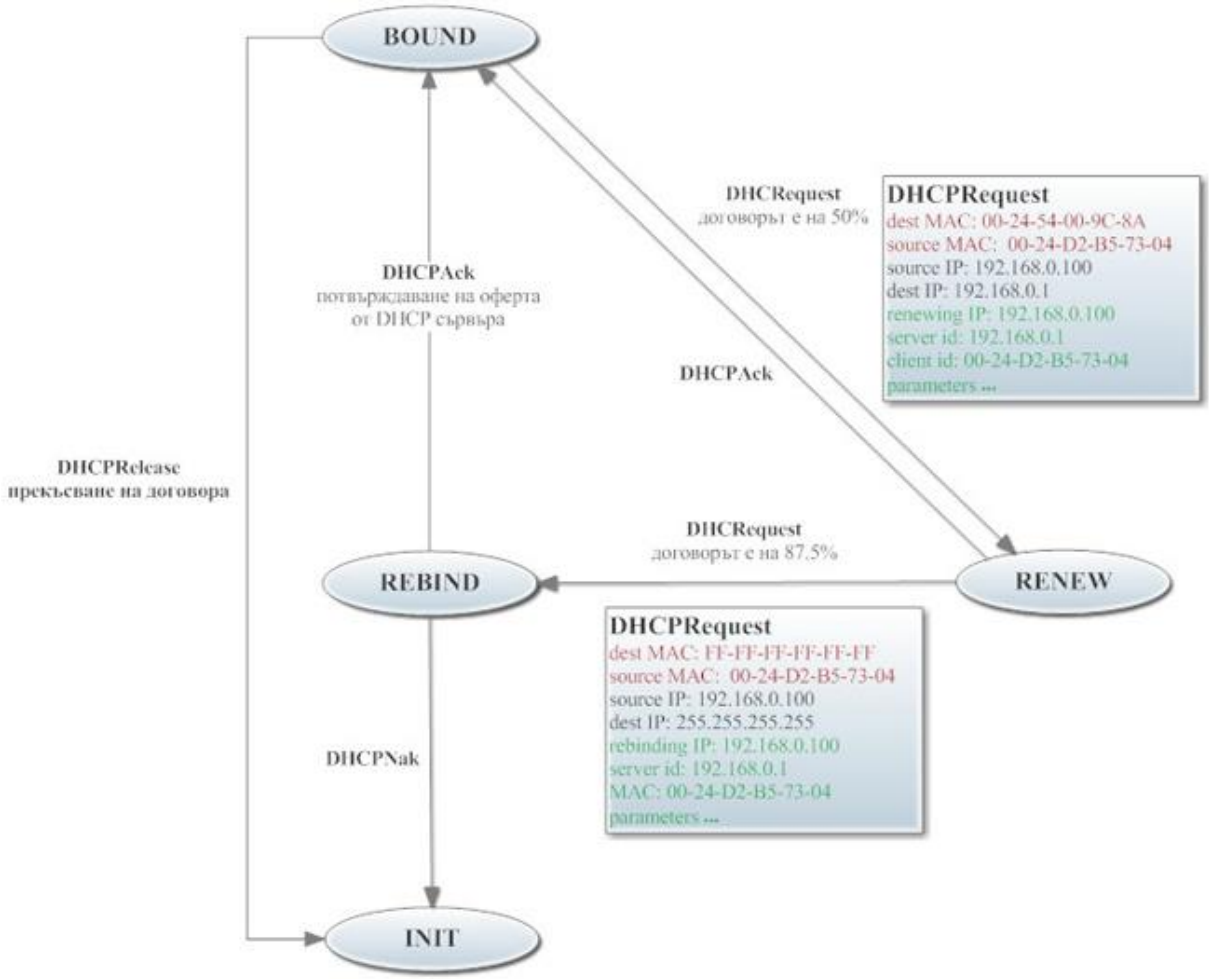
1. Клиентът инициализира ограничена версия на TCP/IP и изпраща DHCPDiscover броадкаст-съобщение за откриване на DHCP сървър. За IP адрес на изпращача се използва 0.0.0.0 (липсва адрес все още), а като адрес на получателя 255.255.255.255 (broadcast) и MAC адрес за локален броадкаст FF-FF-FF-FF-FF-FF. Допълнително е заложен хардуерният адрес и името на DHCP клиента. Статусът на клиента е INIT.
2. DHCP сървърите, получили броадкаст-съобщението, изпращат предложение (DHCPOffer) с IP адрес от диапазона с допустими стойности за раздаване. Предложеният IP адрес временно се резервира, докато сървърът не получи отговор от клиента. Това предположение се изпраща като броадкаст-съобщение (IP 255.255.255.255), включващо и MAC адрес на клиента, изпратил заявката. Статусът на клиента е SELECT.
3. DHCP клиентът отговаря със заявка (DHCPRequest) на първото предложение, при няколко налични. Това съобщение отново се разпространява по мрежата като броадкаст. Получават го всички DHCP сървъри. Тези, на които не е прието предложението връщат резервираните клиентски адреси в списъка на свободните си адреси. Статусът на клиента е REAQUEST.



фигура 168 *Състояния на клиента в процеса на договаряне за IP адрес*

4. Избраният DHCP сървър изпраща потвърждение (DHCPAck) към клиента, приел неговата оферта. Сървърът потвърждава предложения IP адрес и изпраща към клиента необходимата информация за конфигуриране на TCP/IP.
5. Клиентът преминава в състояние BOUND и стартира таймер T1 (Renewal) и таймер T2 (Rebinding). По подразбиране T1 е настроен на 50% от времето на договора, а T2 на 87.5%.

Процесът на подновяване на актуалните параметри и състоянията на клиента са представени на фигура 169.

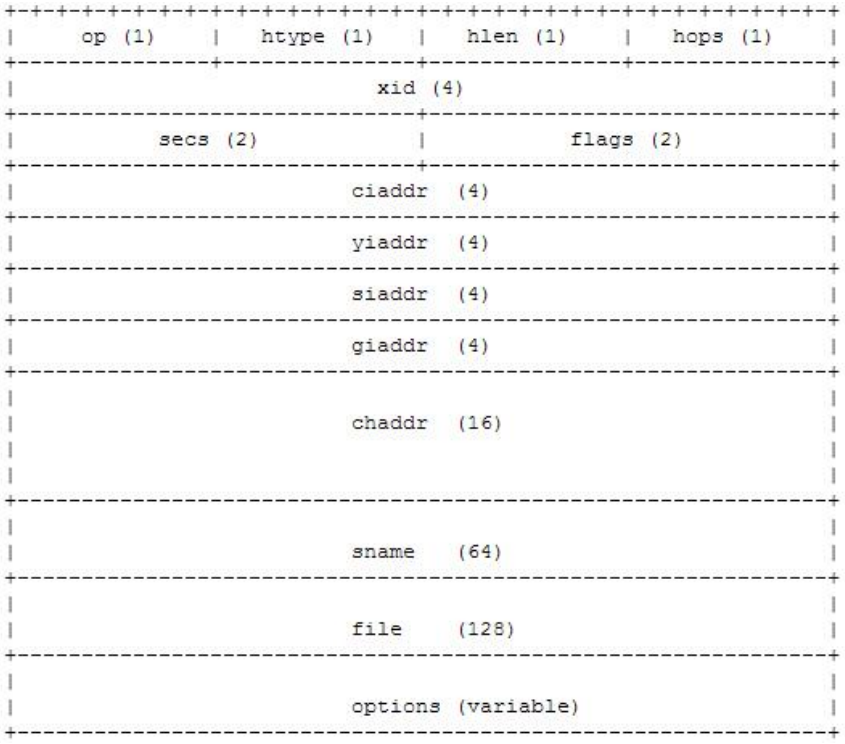


фигура 169 *Състояния на клиента в процеса на подновяване на IP адрес*

1. При изтичане на таймер T1, клиентът навлиза в период на подновяване (Renewal) като се опитва да поднови своя договор с оригиналния DHCP сървър. Клиентът изпраща заявка (DHCPRequest) към DHCP сървъри за подновяване на договора, с максимум три повторения - на 4,8 и 16 секунди. При успешно подновяване DHCP сървърът отговаря с DHCPAck съобщение, съдържащо конфигурационните параметри, които клиентът подновява ако са настъпили промени в досегашните настройки.
2. При липса на отговор от оригиналния DHCP сървър, клиентът продължава функционирането до изтичане на T2 таймера. Клиентът преминава в състояние Rebind и изпраща DHCPRequest броадкаст-съобщение към съществуващи DHCP сървъри за потвърждаване на наличния договор. Ако DHCP сървър потвърди с DHCPAck съобщение се подновява договора. При отрицателен отговор DHCPNak или липса на отговор, клиентът се освобождава

от настройките за IP и рестартира процеса за получаване на нови такива.

Структурата на DHCP съобщението [RFC 2131] е показана на фигура 170.



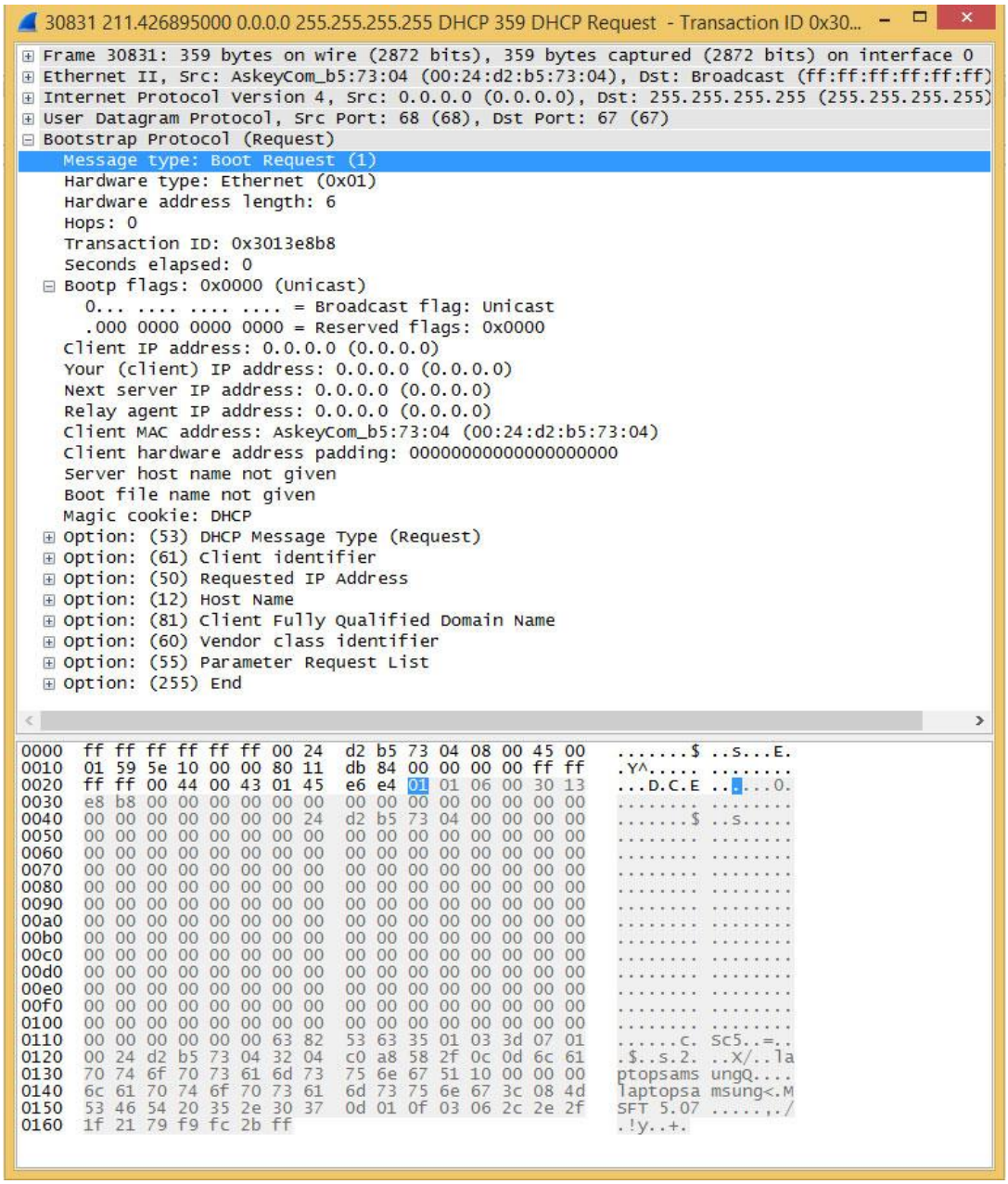
фигура 170 *Структура на DHCP съобщение*

op - 1-байтово поле, идентифициращо типа на DHCP съобщението:
1 = BOOTREQUEST – запитване от клиента;
2 = BOOTREPLY – отговор от сървър.
htype – 1-байтово поле, идентифициращо типа на хардуера (например, Ethernet, Token-Ring или друг тип мрежа). За Ethernet това поле има стойност 1;
hlen – 1-байтово поле, задаващо дължината в байтове на хардуерния адрес;
hops - 1-байтово незадължително поле, задаващо броя на скоковете, които трябва да направи съобщението. Първоначалната стойност зададена от клиента е 0. Шлюзовете увеличават стойността с единица при препращане на DHCP запитването.
xid – 4-байтово поле, задаващо ID на транзакцията. Избира се на случаен принцип от клиента;
secs - 2-байтово поле, задаващо часа, в който е изпратено запитването. Попълва се от клиента;
flags - 2-байтово поле, задаващо типа на съобщението – броадкаст или мултикаст;
ciaddr - 4-байтово поле, съдържащо клиентското IP ако клиентът е в състояния BOUND, RENEW или REBIND и може да отговори на ARP запитване. В противен случай е 0.0.0.0;
yiaddr - 4-байтово поле, съдържащо клиентското IP зададено в DHCP отговора. При DHCP запитване може да приеме стойност:

- 0.0.0.0 - при ciaddr=0.0.0.0;

– IP адресът на клиента, ако има зададен такъв от предишна инициализация.

siaddr - 4-байтово поле, съдържащо нулева стойност или IP адреса на DHCP сървъра, познат на клиента от предишна инициализация;
giaddr- 4-байтово поле, съдържащо нулева стойност или IP адреса на локален шлюз за препращане към отдалечен DHCP сървър;
chadd - 16-байтово поле, съдържащо хардуерния адрес на DHCP клиент;
sname - 64-байтово поле, идентифициращо DHCP сървъра по име. Ако е зададена ненулева стойност може да отговори конкретен DHCP сървър. В противен случай – могат да отговорят всички DHCP сървъри;
file - 128-байтово поле, идентифициращо името на файла за първоначално зареждане, поискано от DHCP клиента;
options – поле с променлива дължина, съдържащо опционални параметри.



фигура 171 *Примерна DHCP request*

Примерна DHCP заявка е показана на фигура 171, където могат да се видят коментираните полета от структурата на DHCP съобщението.

9.5.8. SMTP (Simple Mail Transfer Protocol)

SMTP е протокол за електронна поща, осигуряващ предаване на електронно съобщение между SMTP-подател и SMTP-получател, чрез двупосочно TCP съединение през порт 25. Дефиниран е в RFC 821, а структурата на съобщението в RFC 822. Обменът на съобщения, се реализира чрез последователност от SMTP команди и отговори между изпращача и получателя (фигура 172). Кодовете на командите са четири-символни, като някой от тях са следвани от аргументи. Отговорите представляват цифрови кодове от три позиции с определено значение (таблица 29, таблица 30, таблица 31). Например:

- командата за доставка до пощенска кутия е ***MAIL <SP> FROM <адрес на подателя> <CRLF>***, където <SP> е интервал, а <CRLF> означава начало на нов ред;
- отговорът ***250 OK*** означава успешно изпълнение на подадената команда.

Командите, включени в SMTP протокола, са представени в таблица 28.

команди	значение
HELO	Идентифицира подателя пред получателя
MAIL	Начало на пощенската транзакция
RCPT	Идентифицира получателя на съобщението
DATA	Редовете след тази команда се тълкуват като данни от съобщението до комбинацията <CRLF>. <CRLF>
RSET	Прекратява текущата транзакция като нулира всички буфери и таблици на състоянието
NOOP	Командата се използва за тестване и изисква единствено отговор с OK от получателя.
QUIT	Указва на получателя да затвори връзката след като отговори с OK.
VRFY	Очаква от получателя да потвърди, че аргументът идентифицира потребителя като върне пълното му име и специфицира пощенската му кутия.
SEND	Инициализира mail транзакция за доставка на съобщението до един или повече терминала.
SOML (SEND OR MAIL)	Инициализира mail транзакция за доставка на съобщението до един или повече терминала или пощенски кутии.
SAML (SEND AND MAIL)	Инициализира mail транзакция за доставка на съобщението до един или повече терминала и пощенски кутии.

TURN	Съществуват две възможности за отговор на тази команда: 1. При изпращане на ОК, в отговор, получателят се превръща в SMTP-изпращач или 2. Получателят отказва новата роля и остава като SMTP-приемник.
EXPN	Изисква потвърждение от приемника, че списъка подаден като аргумент идентифицира пощенски кутии. Очакваният резултат е информация за тях.
HELP	SMTP-приемникът връща информация за поддържаните команди.

таблица 28 *SMTP команди*

Отговорите са трицифрени комбинации с йерархична структура, където първата цифра съдържа обща информация, а втората и третата специфицират конкретно възникналата ситуация.

Първа цифра	значение
1bc	Положителен предварителен отговор
2bc	Положителен завършен отговор
3bc	Положителен междинен отговор
4bc	Преходен отрицателен завършен отговор
5bc	Постоянен отрицателен завършен отговор

таблица 29 *Значение на първа цифра*

Втора цифра	значение
a0c	Синтаксис
a1c	Информация
a2c	Връзка
a5c	Пощенска система

таблица 30 *Значение на втора цифра*

Код	значение
211	Отговор за състояние на системата или помощ за системата
214	Помощно съобщение
220 домейн	Услугата е готова
221 домейн	Затваряне на канал за предаване
250	ОК
251	Препращане към цитиран път
354	Стартиране на въвеждане на поща. Край при <CRLF>
421 домейн	Услугата не е достъпна; затваряне на връзка
450	Исканото действие с пощата не е предприето – недостъпна пощенска кутия (заета).
451	Исканото действие е прекъснато – локална грешка в обработката
452	Исканото действие е прекъснато – недостатъчно място за съхранение.
500	Синтактична грешка
501	Синтактична грешка в параметрите
502	Командата не сработва
503	Грешна последователност от команди

504	Командният параметър не може да бъде имплементиран
550	Исканото действие с пощата не е предприето – недостъпна пощенска кутия (не е открита)
551	Потребителят не е локален – опит с път за препращане
552	Прекъснато действие – препълнено пространство за съхранение
553	Непредприето действие – липсва име на пощенската кутия
554	Неуспешна транзакция

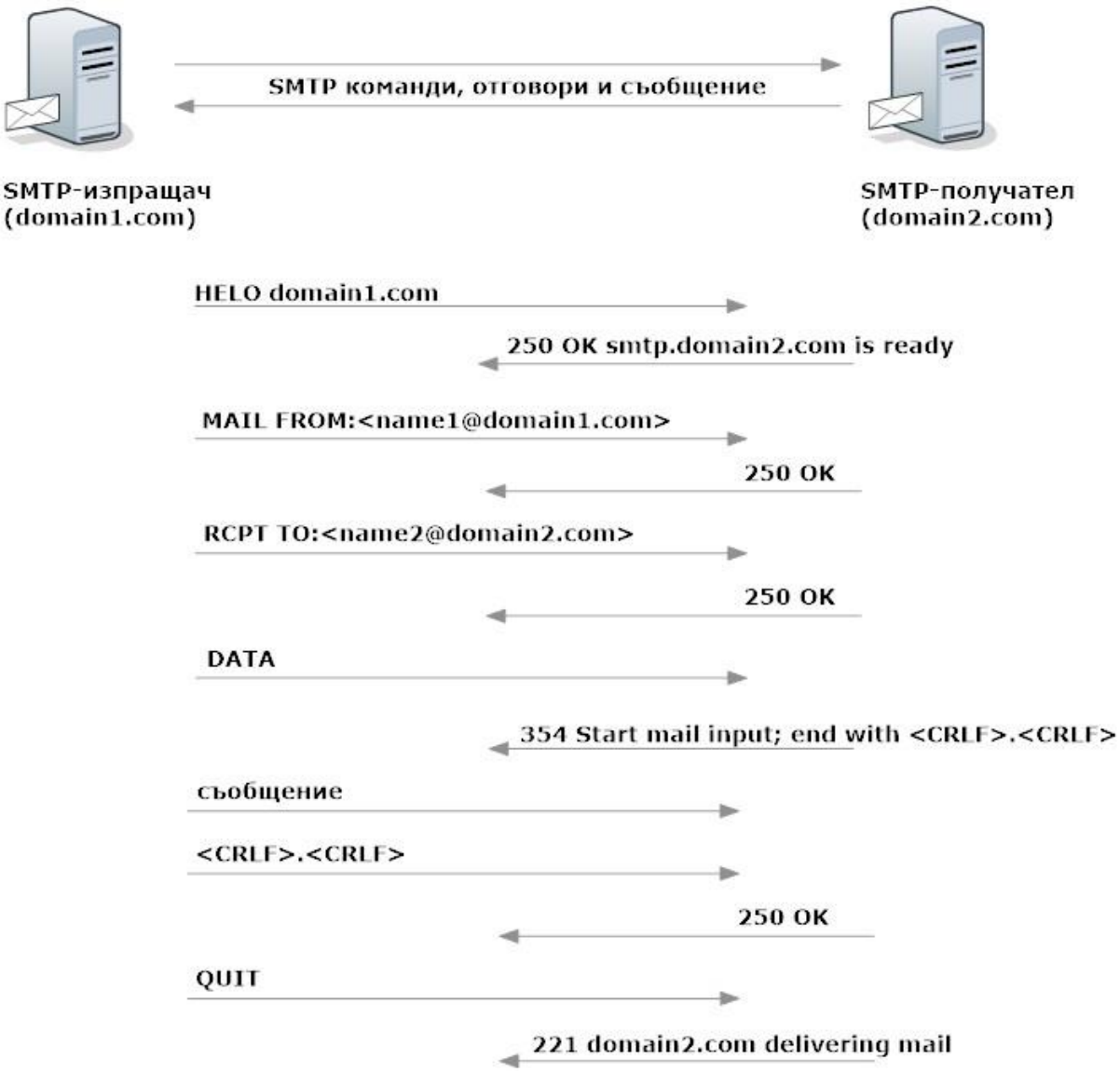
таблица 31 *Отговори*

На фигура 172 е показана директна комуникация между два SMTP сървъра за успешно предаване на e-mail съобщение между подател и получател.

Самото съобщение се състои от:

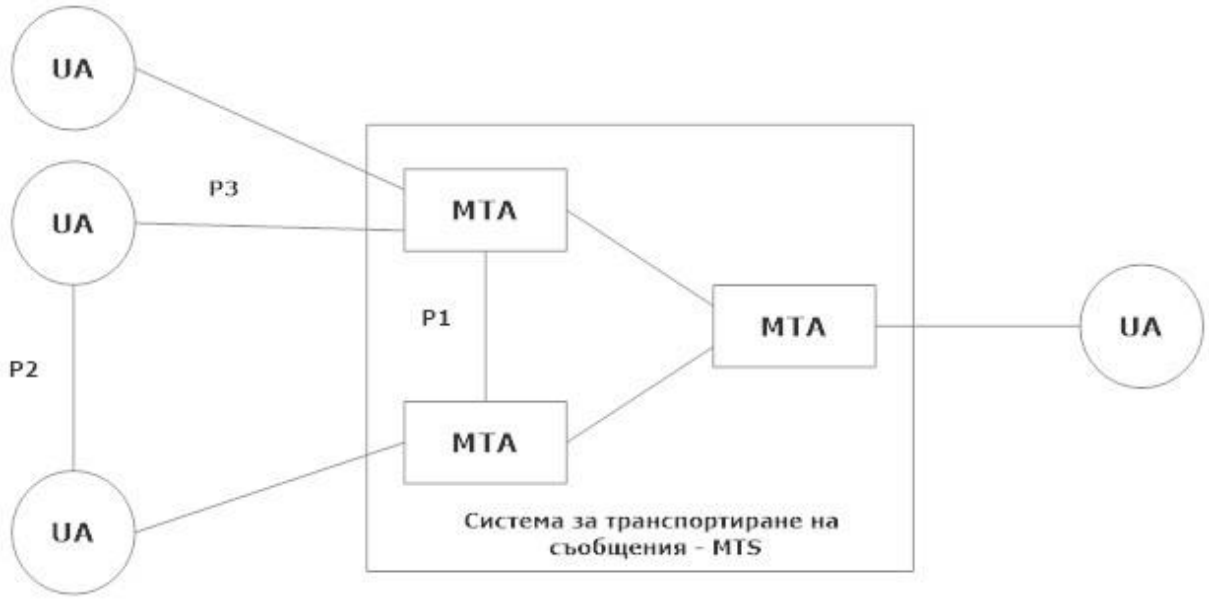
- заглавна част (header) – необходима за обработката и доставката на съобщението. Съдържа полета, някои от които имат определен формат (например-*From:*), а други не (например-*Subject:*). Задължителни са само част от полетата – *Date:, From:, To:* или *Cc:*;
- тяло (body) – съдържащо самото съобщение. Първоначално, тялото е можело да пренася само английски текст, кодиран със 7-битов US ASCII код. По-късно, масовото използване на електронната поща налага разработването на стандарта MIME (Multipurpose Internet Mail Extension), който разширява възможностите на SMTP протокола като премахва:

- невъзможността за предаване на текст, съдържащ символи, които не могат да се кодират със 7-битовия US ASCII код;
- ограничението за максимална дължина на съобщението;
- невъзможността SMTP-шлюзове да обработват нетекстови съобщения;
- и други.



фигура 172 *Директна комуникация между два SMTP сървъра*

За обработването и трансфера на съобщенията се грижат потребителски агенти (UA-user agent) и агенти за транспортиране на съобщения (MTA-message transfer agent), базирани на модела за MHS (Message Handling System) система, като част от препоръките CCITT X.400 (фигура 173). Потребителските агенти са приложен софтуер, осигуряващ интерфейс между потребителя и MHS системата за обмен на съобщения. Те се свързват с MTA агентите за осъществяване на предаването, които от своя страна извършват действия като приемане, маршрутизиране, изпращане на съобщението от/към други MTA или UA агенти, уведомяване за доставено или недоставено съобщение. Пример за UA агенти могат да бъдат e-mail клиентите (например, Windows Live Mail), а SMTP сървърните програми са MTA агенти (например, Sendmail за UNIX).



фигура 173 *MHS система*

Обозначенията P1(X.411), P2(X.420) и P3(X.410) се отнасят за три ключови протокола, свързани с маршрутизирането на съобщения, специфичните услуги за потребителите и въздействие върху параметрите на MTA агентите.

P1 е протокол, който дефинира полетата за пакетиране на изпращаните данни и определя начина на маршрутизиране на съобщенията между два MTA агента. Информацията, която е заложена в тези полета определя идентификатор на съобщението (MI), информация за потребителя, инструкции за доставянето му, обратна информация.

Протоколът P2 е насочен към улесняване на потребителите като им осигурява услуги, свързани с изпращане на съобщения до конкретни потребители, получаване на информация за предмета на съобщението и определяне на вида на известието, очаквано от получателя.

P3 е протокол, определящ правилата за комуникация с MTA агентите. Например, чрез този протокол потребителят може да промени параметри като необходимост от парола, максимално допустим размер на съобщението и др.

9.5.9. HTTP (Hypertext Transfer Protocol)

HTTP е протокол за трансфер на хипертекст. Терминът hyper означава, че документът съдържа връзки, които могат да се избират. Резултатът довежда до формирането на Световната уеб мрежа. Неговото

развитие осигурява поддържането на сложни типове данни, които лежат в основата на съвременния WEB.

HTTP работи в приложния слой на TCP/IP стека. Използва TCP за гарантирана доставка, стандартен сървърен порт 80 и следните компоненти, при своето функциониране:

- Потребителски агент (UA – user agent) – софтуер, инициализиращ заявката. Може да бъде браузер, паяк или друг потребителски инструмент;
- Ресурси – обект или услуга, идентифицираща се чрез URI (Uniform Resource Identifiers);
- Първоначален сървър – HTTP сървър, хостващ ресурсите;
- Прокси – посредничи в обмена на информация между клиента и оригиналния сървър. Работи като клиент и сървър. Заявката може да се обслужи от него локално или да се прехвърли към друг сървър.

Прозрачно прокси – не модифицира заявката или отговора.

Непрозрачно прокси – модифицира заявката с цел добавяне на допълнителна услуга.

Прокси сървърите кешират WEB страниците, които използват по-късно в отговор на заявки.

- Шлюз – получаващ агент, работещ като слой над други сървъри и при необходимост транслира заявката към подразбиращия се протокол;
- Тунел – софтуерен интерфейс-посредник, осигуряващ прозрачно препредаване на съобщение между две връзки (например, преодоляване на firewall);
- Комуникационна верига – виртуална верига за осъществяване на комуникацията между две приложения;

Най-често инициализирането на една HTTP сесия се осъществява от потребителския агент и може да протече по няколко варианта:

- UA да комуникира директно с оригиналния сървър;

фигура 174 *Директна комуникация*

- Комуникацията да премине през посреднически устройства прокси, шлюз или тунел;

фигура 175 *Комуникация чрез посреднически устройства*

- Посредническо устройство да отговори на заявката като използва кеша си (получава се при нетунелиране на връзката).

фигура 176 *Използване на кеширана заявка*

Основните съобщения за комуникация са заявка и отговор (таблица 32).

HTTP-message *= Request | Response; HTTP/1.1 messages*

Форматът им спазва следната структура:

generic-message = start-line

**(message-header CRLF)*

CRLF

[message-body],

където:

start-line **=** *Request-Line | Status-Line*

Request	Response
Request = Request-Line **((general-header* *| request-header* *| entity-header) CRLF)* *CRLF* *[message-body]*	*Response = Status-Line* **((general-header* *| response-header* *| entity-header) CRLF)* *CRLF* *[message-body]*

таблица 32 *Основни съобщения*

**rule* - повторение. Пълният вариант на формата е *"<n>*<m>element"*, където <n> е най-малко, а <m> - най-много. Стойността по подразбиране са 0 и безкрайност, което означава, че:

- *"*(element)"* – липсва или се повтаря определени пъти;
- *"1*element"* – най-малко един път;
- *"1*2element"* – един или два пъти.

Използвани съкращения:
LWS = [CRLF] 1*(SP | HT)
CRLF = CR LF
OCTET = <всяка 8-битова поредица от данни>
CHAR = <всеки US-ASCII символ (octets 0 - 127)>
UPALPHA = < всяка US-ASCII главна буква "A".."Z">
LOALPHA = <всяка US-ASCII малка буква "a".."z">
ALPHA = UPALPHA | LOALPHA
DIGIT = <всяка US-ASCII цифра "0".."9">
CTL = <всеки US-ASCII контролен символ (octets 0 - 31) и DEL (127)>
CR = <US-ASCII CR, carriage return (13)>
LF = <US-ASCII LF, linefeed (10)>
SP = <US-ASCII SP, space (32)>
HT = <US-ASCII HT, horizontal-tab (9)>
<"> = <US-ASCII double-quote mark (34)>

Request-Line **=** *Method SP Request-URI SP HTTP-Version CRLF*

Дефинираните методи са:

- "OPTIONS" – подава заявка за информация относно параметрите за комуникация. Отговорът не се кешира.
- "GET" – извлича информацията от указания URI ресурс
- "HEAD" – методът е подобен на „GET“, но отговорът не включва тяло на съобщението (message-body), а само метаинформация включена в заглавната част на единицата.

- "POST" – използва се за достъп до определена самостоятелна единица, подчинена на актуалния URI за изпращане на данни към сървъра.
- "PUT" - заменя всички настоящи представяния на целевия ресурс с каченото съдържание
- "DELETE" – подава заявка за изтриване на всички представяния за посочения URI ресурс
- "TRACE" – изпраща loop-back съобщение към посочения ресурс
- "CONNECT" - създава тунел към сървъра идентифициран от даден URI.

***Status-Line** = HTTP-Version SP Status-Code SP Reason-Phrase CRLF*

Статус кодовете са трицифрени числа с категоризираща първа цифра:

- 1xx: Informational – получена заявка и продължаващ процес;
- 2xx: Success – успешно получаване на действията, разбиране и реализиране;
- 3xx: Redirection – предприемане на по-нататъшни действия за завършване на заявката;
- 4xx: Client Error – грешен синтаксис или невъзможно изпълнение;
- 5xx: Server Error – неизпълнение на заявката от страна на сървъра.

Message-header на съобщението включва general-header, request-header, response-header, entity-header. Всяка от тях е във формат:

message-header = field-name ":" [field-value]

- general-header – съдържа полета, които са общи за двата типа съобщения (заявка и отговор) и се прилагат само към съобщението;

Cache-Control – включва директиви, които контролират кеширането при UA клиентите, посредниците и оригиналните сървъри;
Connection – характеризира типа на връзката;
Date – съдържа датата и часа на запитването или отговора;
Pragma – посочва директивите, които могат да се приложат за всеки получател по веригата на отговора или заявката;
Trailer – посочва, че дадено множество от полета на хедъра ще бъдат в трейлъра на съобщението, кодирано на парчета за трансфер (големите потоци се разбиват на парчета - chunks);
Transfer-Encoding – посочва използвания метод за трансфер и кодиране на тялото на съобщението;
Upgrade – указва допълнителните протоколи поддържани от клиента и биха могли да се използват за съвместимост със сървъра,
Via – използва се от посредническите устройства (шлюзове и проксита) за проследяване на препращаните съобщения и използваните протоколи;
Warning – служи за пренос на допълнителна информация относно статуса и трансформацията на съобщението.

• request-header – пренася допълнителна информация, която не се съдържа в стартовия ред или общия хедър. Полетата са следните:

Accept – специфицира типовете медия, допустими за включване в отговора;
Accept-Charset – определя кодовата таблица на отговора;
Accept-Encoding – указва възможните типове кодиране за съдържанието;
Accept-Language – определя възможните естествени езици за използване;
Authorization – използва се за автентикация на UA;
Expect – специфицира изисквания към сървъра, очаквани от клиента;
From – указва email адреса на потребителя, който използва и контролира UA агента;
Host – хоста и порта на изисквания ресурс;
If-Match – посочва получените, от по-ранна комуникация с този ресурс, единици, които са актуални към момента;
If-Modified-Since – съдържа датата, след която се проверява за актуализация на изисквания ресурс т.е. за проверка на актуалността на кеша.
If-None-Match – условно задава на сървъра да изпълни искания метод само ако една от изброените стойности в този таг се съдържа в таговете за единици в полето Etag;
If-Range – служи за опресняване на единиците в кеша в зависимост от датата;
If-Unmodified-Since – указва на сървъра да изпълни операцията ако ресурса не е променян от посочената дата;
Max-Forwards - въвежда ограничение за броя на препращанията през шлюзове и проксита;
Proxy-Authorization – използва се при идентифициране на клиента пред прокси;
Range – определя част от единицата;
Referer – указва на сървъра URI адреса на ресурса от където е получена заявката;
TE – свързано е с указване на кодировката при трансфер;
User-Agent – съдържа информация за агента, който генерира заявката.

• response-header – пренасят допълнителна информация, която не се съдържа в стартовия ред или общия хедър.

Accept-Ranges – дефинира частта от ресурса, която трябва да се приеме като отговор;
Age- неотрицателно десетично число, показващо времето в секунди от последното генериране на заявката (при кеширана такава);
ETag – пренася името на тага на единицата (entity tag), участваща в отговора;
Location – използва се за пренасочване на получателя към ресурс различен от заложеният в заявката;
Proxy-Authenticate – указва схемата и параметрите за автентикация към прокси сървъра. Свързано е с отговор 407 (Proxy Authentication Required);

Retry-After – използва се с отговор 503 (Service Unavailable) или 3xx (Redirection) и посочва минималното време на изчакване на потребителския агент преди изпозването на пренасочена връзка. Формат:

Retry-After = "Retry-After" ":" (HTTP-date | delta-seconds)

Server – задава информация за софтуера на оригиналния сървър, обработващ заявката;

Vary – указва, че единицата има няколко представяния и може да варира спрямо специфичните изисквания на заявката;

WWW-Authenticate – включва се при отговор 401 (Unauthorized) и указва схемата и параметрите за автентикация;

- entity-header – съдържа информация за тялото на единицата или ресурса от заявката.

Allow – задава множеството от методи поддържани от ресурса, идентифициращ се с използваното URI;

Content-Encoding – указва използвания тип за пренос;

Content-Language – определя естествения език на аудиторията, за която е предназначена единицата;

Content-Length – задава размера на тялото на единицата в октети;

Content-Location – задава ресурсното местоположение на единицата;

Content-MD5 – проверка на целостта на съобщението чрез MD5 резюме;

Content-Range – придружава част от съдържанието на единицата и указва неговото местоположение;

Content-Type – специфицира типа медия на тялото;

Expires – показва датата/часа точно преди изтичане на отговора;

Last-Modified – последната дата/час на модифициране на ресурса.

Message-body на съобщението съдържа информацията, която трябва да се покаже в браузъра.

Http схемата, която се използва за достъпване на мрежов ресурс е във вида:

http_URL = "http:" "//" host [":" port] [abs_path ["?" query]]

Допълнителна информация за протокола може да се получи от документ RFC 2616.

9.5.10. FTP (File Transfer Protocol)

FTP е протокол за обмен на файлове между хостове, използвайки услугите на TCP. Основната му употреба е свързана с трансфер на файлове между хостове в Internet. Нивата на достъп на всеки потребител се

определят от FTP сървъра, чрез задаване на потребителско име и парола. Подробно описание на FTP се намира в документ RFC 959.

9.5.11. DNS (Domain Name System)

DNS е система за управление на имената в Интернет и има йерархична дървовидна структура. Основното и предназначение е да асоциира IP адреси с буквено-цифрови имена, което позволява хостовете да бъдат групирани по географски принцип или по тяхната принадлежност към някаква организация.

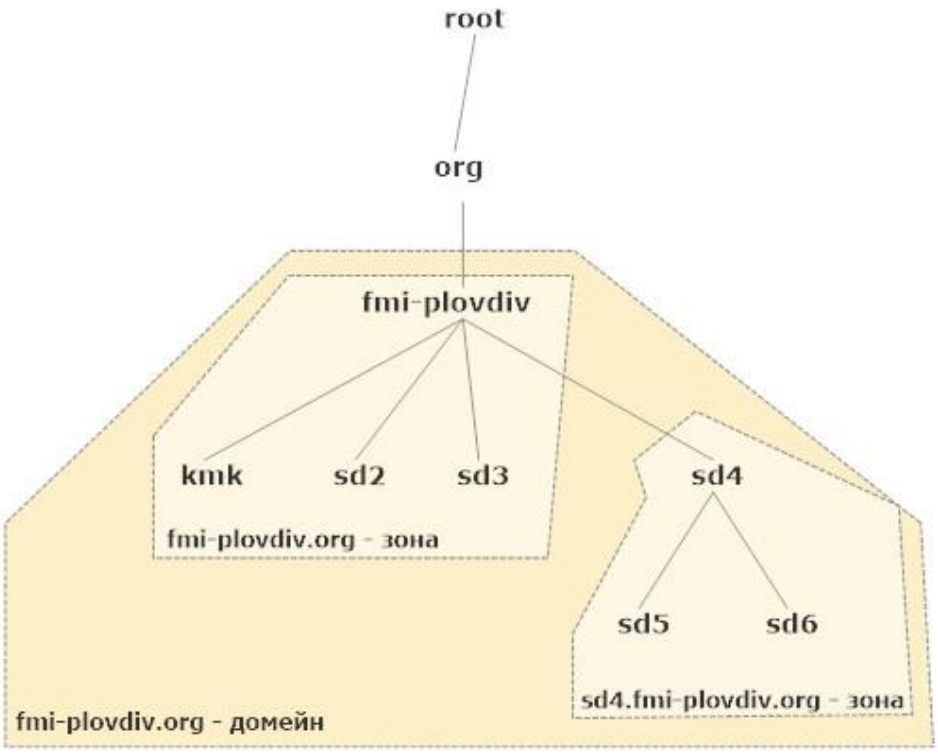
Имената се състоят от отделни части разделени с точка, като най-дясната част представлява име на област, която е най-високо в йерархията (top-level domain). Например, трибуквените означения, посочени по-долу, означават области от определен тип, а двубуквените са за обозначаване на държави.

- edu - образователни институции;
- gov - правителствени организации;
- mil - военни организации;
- com - големи корпорации или бизнес организации;
- net - доставчици на Интернет услуги;
- org - други видове организации;
- au – Австралия;
- bg – България.

Втората част от името (отдясно-наляво) се използва за идентифициране на организация, а всяка следваща част може да бъде име на по-малко подразделение, име на компютър или устройство. Например, kmk.fmi-plovdiv.org може да означава компютър, предлагащ услугата WWW в подразделение kmk на организацията fmi-plovdiv.

DNS позволява пространството от имена да бъде разделено на зони, съдържащи информация за един или повече DNS домейни. За всеки включен в зоната домейн тя се явява авторитетен източник на информация. Първоначално зоната се явява авторитетна база данни за едно DNS име. При добавяне на поддомейни към това име те могат да станат част от същата зона или да обособят нова такава. На фигура 177 е обособена примерна зона за домейна fmi-plovdiv.org, включваща и управление на поддомейните kmk, sd2, sd3. Отделно за управлението на поддомейна sd4 е

обособена втора зона. В този случай първата зона трябва да съдържа няколко RR (resource records) записа, които да пренасочват към авторитетни DNS сървъри за втората зона.



фигура 177 *Домейни и зони*

Домейн – възел в DNS дървото, включващ всички възли под него;
Зона – част от дървото, за която даден DNS сървър е авторитетен (притежава необходимата информация за преобразуване). Може да включва един или няколко домейна;
Асоцииране (mapping) и преобразуване (resolving) – откриване на IP адрес по името на домейн;
Обратна асоциация (inverse mapping) – открива домейн по IP адрес.

За преобразуването и съответствието между имена и IP адреси се грижат специализирани компютри, наречени DNS сървъри (сървъри на имената). За всяка една зона в Интернет трябва да има поне един сървър за имена, който да е авторитетен. По този начин цялата йерархия от имена в Интернет се реализира чрез йерархията от сървъри на имената. Тези сървъри съдържат записи на ресурси (RR), които позволяват асоциирането на име с IP адрес.

Примери за RR записи:

- тип A - асоциира името на хоста с IP адрес

```
mycomputer.mydomain.net      IN  A  192.168.55.10
```

В примера mycomputer.mydomain.net сочи към IP адрес 192.168.55.10.

- тип CNAME - пренасочва дефинираната област към домейна

```
mydomain.net      IN   A   192.168.55.2

test.mydomain.net      CNAME   mydomain.net
```

В примера заявката към test.mydomain.net се пренасочва към mydomain.net

- тип NS - посочва авторитетен сървър за домейна

```
mydomain.net      IN   NS   nameserver1.mydomain.net
```

В примера nameserver1.mydomain.net се явява авторитетен за mydomain.net

- тип MX - посочва сървър за обработка на пощенските съобщения. Могат да са повече сървъра подредени в последователност чрез посочване на номер.

```
mydomain.net      MX   1   mail1.mydomain.net
```

В примера mail1.mydomain.net отговаря за пощенските съобщения на посочения домейн

Поддържат се първостепенни и второстепенни DNS сървъри, където вторите получават информация от първите на определен период от време (например, около 3 часа).

Децентрализацията на базата данни решава проблеми като:

- единична точка за отказ;
- обем на трафика;
- далечна централизирана база данни;
- поддръжка.

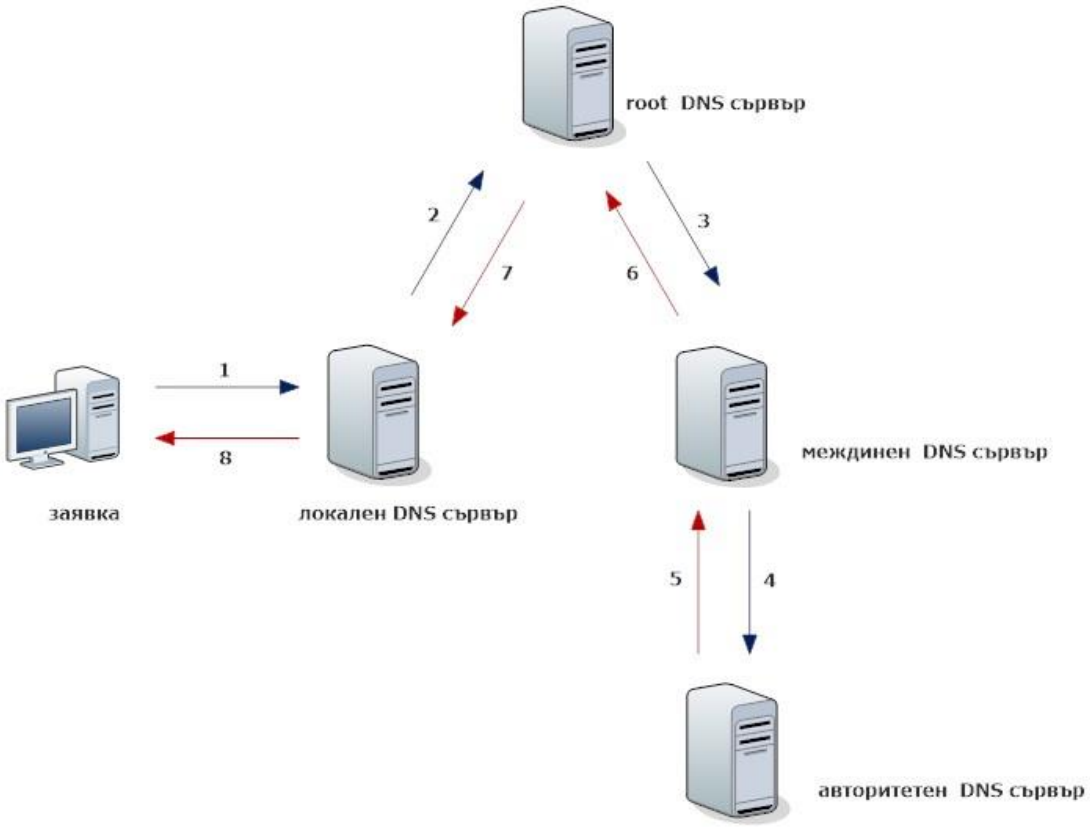
Този подход позволява използването на локални DNS сървъри (default name server) от ISP компаниите, през които да преминават локалните DNS заявки.

В някои случаи определени сървъри може да не разполагат с необходимата информация (в този случай не се считат за авторитетни) и затова се налага обръщение към друг сървър за имена с цел преобразуване на заявката. Този процес на препредаване преминава през сървърите за имена от най-високо ниво, което позволява обхождането на дървото от имена до намиране на подходящия сървър. Преминава се през следните етапи:

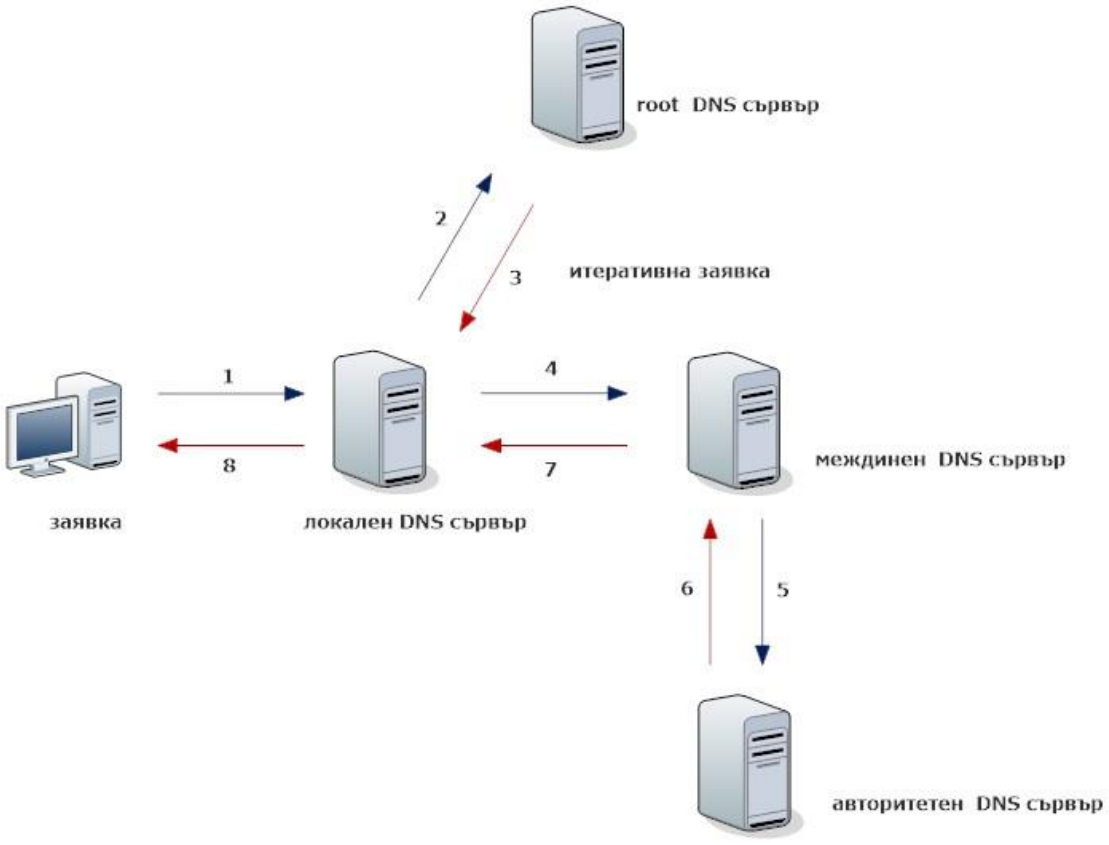
1. При получена заявка DNS сървърът (по подразбиране) проверява дали притежава това име в зоната, която управлява и ако го намери отговаря на заявката;
2. При липса на името се проверява кеша (съдържа информация за около два дни) и ако го открие там го връща в отговор на заявката като включва уведомяване откъде е получена информацията.
3. Ако името липсва и в кеша, сървърът се превръща в клиент и изпраща заявка към авторитетен сървър за отговор. Действието по препращането може да се повтори многократно (рекурсия).

Рекурсията е процес, даващ възможност на всеки DNS сървър да намери сървърите, които са авторитетни за отделните имена (фигура 178). Заявката е от типа server-to-server query. По подразбиране DNS клиентите са настроени да изискват рекурсия от DNS сървърите, които най-често са конфигурирани да я подкрепят.

Итерацията е процес на повторение на заявката от DNS клиента към различни DNS сървъри за намиране на авторитетен сървър (фигура 179). Заявките са от тип client-to-server query. Итерацията може да е следствие от забрана на рекурсията в DNS сървъра или самият клиент не е настроен за използването на такава.

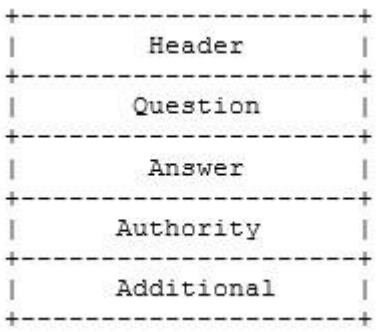


фигура 178 *Заявка към авторитетен сървър с използване на междинен сървър (рекурсия)*

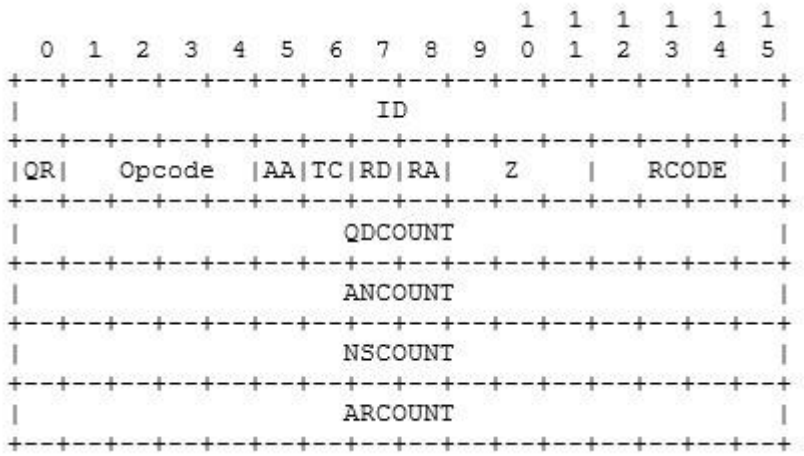


фигура 179 *Заявка към авторитетен сървър чрез итеративна заявка*

Форматът на DNS съобщението е представен (RFC 1035) на фигура 180 и фигура 181. В основната си част има 5 раздела, които се използват в зависимост от типа на съобщението. Хедърът присъства във всяко съобщение.



фигура 180 *Формат на DNS съобщението*



фигура 181 *Формат на хедъра на съобщението*

ID – 16-битово поле за идентификация между заявка и отговор;
QR – поле от 1 бит, определящо съобщението като заявка или отговор:

- QR=1 – отговор;
- QR=0 – заявка.

Opcode – 4-битово поле, което доопределя QR:

- 0 – стандартна заявка (QUERY);
- 1 – обратна заявка (IQUERY);
- 2 – запитване за състоянието на сървъра (STATUS).

AA – 1-битово поле, определящо дали сървърът е авторитетен (стойност 1) за домейна във въпроса;
TC - 1-битово поле, сигнализиращо при 1, че отговорът надхвърля 512 байта и е отрязан до тях;
RD - 1-битово поле, съдържащо 1 при желана рекурсия;
RA - 1-битово поле, информиращо за достъпна рекурсия. Стойността му е 1 ако сървърът поддържа рекурсия;

Z – винаги носи нулева стойност;
RCODE – 4-битово поле, което е част от кода на отговора със следните стойности:

- 0 – липсва грешка;
- 1 – грешен формат;
- 2 – пропадане на сървъра;
- 3 – авторитетен сървър сигнализира за грешно име на домейн;
- 4 – сървърът не поддържа такъв тип заявка;
- 5 – липса на поддръжка на такъв тип заявка;
- 6-15 – запазени за бъдещо използване.

QDCOUNT – указва броя на въпросите в секция Question;
ANCOUNT - указва броя на RR записите в секцията Answer;
NSCOUNT – указва броя на NSRR (name server resource records) в секцията Authority;
ARCOUNT - указва броя на RR записите в секцията Additional;

БИБЛИОГРАФИЯ

[1] A Simula 67 bibliography, http://folk.uio.no/simula67/bibliography.shtml (последно посетен на 18.03.2013)

[2] Aaron Balchunas, "Open Shortest Path First", http://www.routeralley.com/ra/docs/ospf.pdf (последно посетен на 18.03.2013)

[3] Aaron Balchunas, "Routing Information Protocol", http://www.routeralley.com/ra/docs/rip.pdf (последно посетен на 18.03.2013)

[4] Abbas Jamalipour, „Broadband ISDN: Architecture and Protocols", http://iwayan.info/Lecture/ISDN_S1/chap14-15b_BISDN_Protocol.PDF (последно посетен на 18.03.2013)

[5] Alan Alberecht, Patricia Thaler, "Introduction to 100VG-AnyLAN and the IEEE 802.12 Local Area Network Standard", http://www.hpl.hp.com/hpjournal/95aug/aug95a1.pdf (последно посетен на 18.03.2013)

[6] ATM Protocol, http://www-ee.uta.edu/online/Wang/Atm-Protocol.pdf (последно посетен на 18.03.2013)

[7] ATM Technology, http://docstore.mik.ua/univercd/cc/td/doc/product/atm/l2020/2020r211/sysover/atmtech.htm (последно посетен на 18.03.2013)

[8] Automatic Repeat Request, http://www.doc.ic.ac.uk/~pjm/nac/lecture_datalink.pdf (последно посетен на 18.03.2013)

[9] Avi Kak, „Public-Key Cryptography and the RSA Algorithm" https://engineering.purdue.edu/kak/compsec/NewLectures/Lecture12.pdf (последно посетен на 2.03.2014)

[10] Ayushi, „A Symmetric Key Cryptographic Algorithm", http://www.ijcaonline.org/journal/number15/pxc387502.pdf (последно посетен на 1.03.2014)

[11] BASIC CABLE TV BACKGROUND: MODULATION SIGNAL FORMATS AND COAXIAL CABLE SYSTEMS, http://catalogue.pearsoned.co.uk/samplechapter/0130864218.pdf (последно посетен на 18.03.2013)

[12] Boson, "NetSim8 User Manual", http://www.boson.com/Files/Support/NetSim-8-User-Manual.pdf (последно посетен на 18.03.2013)

[13] Brandon Provolt, „xDSL Tutorial", http://www.schottcorp.com/news/technical_papers/xDSL%20Tutorial.pdf (последно посетен на 18.03.2013)

[14] Capt. Noël Davis, Capt. Scot Ransbottom and Lt.Col. Drew Hamilton, „Teaching Computer Networks Through Modeling", Electrical Engineering and Computer Science, United States Military Academy,West Point, New York 10996

[15] Carlos J. Bernardos, Alberto García-Martínez, Celeste Durán, „Computer networking teaching experiences using COTS routers and virtual environments: the UC3M laboratory", http://www.it.uc3m.es/cjbc/papers/bernardos_uvil.pdf (последно посетен на 18.03.2013)

[16] CELLSoft, "ATM - Concepts and Architecture", http://www.cellsoft.de/telecom/atmconcepts.htm (последно посетен на 18.03.2013)

[17] Charles M. Kozierok, "DHCP Lease "Life Cycle" Overview (Allocation, Reallocation, Renewal, Rebinding and Release) and Lease Timers",

http://www.tcpipguide.com/free/t_DHCPLeaseLifeCycleOverviewAllocationReallocationRe.htm (последно посетен на 18.03.2013)
[18] Circuit and Packet Switching, http://klamath.stanford.edu/~molinero/thesis/chapter.2.pdf (последно посетен на 18.03.2013)
[19] Cisco Systems, „Introduction to xDSLTechnology“, ftp://ftp-eng.cisco.com/cons/workshops/isp-workshop/StudentCD-Rev-E/Networkers99/203.pdf (последно посетен на 18.03.2013)
[20] Cisco, “Border Gateway Protocol (BGP)” http://www.cisco.com/en/US/tech/tk365/tk80/tsd_technology_support_sub-protocol_home.html (последно посетен на 18.03.2013)
[21] Cisco, “How LAN Switches Work”, http://www.cisco.com/en/US/tech/tk389/tk689/technologies_tech_note09186a00800a7af3.shtml (последно посетен на 18.03.2013)
[22] Cisco, “How NAT Works”, http://www.cisco.com/en/US/tech/tk648/tk361/technologies_tech_note09186a0080094831.shtml (последно посетен на 18.03.2013)
[23] Cisco, “Open Shortest Path First (OSPF)”, http://www.cisco.com/en/US/tech/tk365/tk480/tsd_technology_support_sub-protocol_home.html (последно посетен на 18.03.2013)
[24] CISCO, “Troubleshooting Fiber Distributed Data Interface”, http://www.cisco.com/en/US/docs/internetworking/troubleshooting/guide/tr1905.html (последно посетен на 18.03.2013)
[25] D. Meyer, “Administratively Scoped IP Multicast”, http://www.ietf.org/rfc/rfc2365.txt (последно посетен на 18.03.2013)
[26] David Haccoun, Samuel Pierre, “Automatic Repeat Request”, http://radio-1.ee.dal.ca/~ilow/4540/readings/crc_ch14.pdf (последно посетен на 18.03.2013)
[27] Deep Medhi, „Routing Management in the PSTN and the Internet: A Historical Perspective“, Journal of Network and Systems Management, vol. 15, no. 4, December 2007, http://sce.umkc.edu/~dmedhi/papers/m-jnsm-07.pdf (последно посетен на 18.03.2013)
[28] Deloadvert.com, “Моделирование”, http://www.compmodel.ru/394/ (последно посетен на 18.03.2013)
[29] Douglas Comer, “Computer Networks and Internets, 5e”, http://netbook.cs.purdue.edu/ (последно посетен на 18.03.2013)
[30] DWDM-Dense wavelength division multiplexing, http://www.ee.columbia.edu/~bbathula/courses/HPCN/chap04_part-3.pdf (последно посетен на 18.03.2013)
[31] Farid Farahmand, Qiong (Jo) Zhang, “Circuit Switching”, http://web.ccsu.edu/technology/farahmand/ccsu/courses/cet543/resources/ch69_circuit_switcing.pdf (последно посетен на 18.03.2013)
[32] Frequency division multiplex, http://www.d.umn.edu/~ibra0130/a1-09a.pdf (последно посетен на 18.03.2013)
[33] FREQUENCY DIVISION MULTIPLEXING FOR ANALOGUE COMMUNICATIONS, http://www.crg.cs.nott.ac.uk/~mpc/craven_phd_chap3.pdf (последно посетен на 18.03.2013)
[34] Gary Audin, „Can the PSTN be Shut Down?“, http://www.webtorials.com/main/resource/papers/delphi/paper9/pstn_shutdown.pdf (последно посетен на 18.03.2013)
[35] General Telecom, “Ethernet Frame Types”, http://telecom.tbi.net/frmlan.html, (последно посетен на 18.03.2013)

[36] Gerald P. Ryan, "Dense Wavelength Division Multiplexing", CIENA Corporation, https://aresu.dsi.cnrs.fr/IMG/pdf/dwdm_ciena.pdf (последно посетен на 18.03.2013)
[37] Greg Watson, Alan Albrecht, Joe Curico, Dan Dove, Steve Goody, John Grinham, Michael Spratt, Pat Thaler, "The Demand Priority MAC Protocol", http://www.hpl.hp.com/techreports/94/HPL-94-58.pdf (последно посетен на 18.03.2013)
[38] GregWatson, MartMolle, "100 Base-T/IEEE 802.12/ Packet Switching", http://www.hpl.hp.com/techreports/96/HPL-96-58.pdf (последно посетен на 18.03.2013)
[39] Harry Perros, "Connection-Oriented Networks", http://www4.ncsu.edu/~hp/Chapter2.pdf (последно посетен на 18.03.2013)
[40] Hedrick C., "Routing Information Protocol", Request for Comments: 1058, http://www.ietf.org/rfc/rfc1058.txt (последно посетен на 18.03.2013)
[41] HP, "100VG-AnyLAN", ftp://ftp.hp.com/pub/networking/software/59636588.pdf (последно посетен на 18.03.2013)
[42] IBM Corporation , "802.2 LLC Frame", http://publib.boulder.ibm.com/infocenter/zvm/v5r4/index.jsp?topic=/com.ibm.zvm.v54.kdpl0/hcsk3b31234.htm (последно посетен на 18.03.2013)
[43] IEEE Standards Association, http://standards.ieee.org (последно посетен на 18.03.2013)
[44] Introduction to cryptography, http://vig.prenhall.com/samplechapter/0130614661.pdf (последно посетен на 2.03.2014)
[45] Introduction to Using OPNET Modeler, http://www.sce.carleton.ca/faculty/lambadaris/courses/5001/opent_tutorial.pdf (последно посетен на 18.03.2013)
[46] J. Theunis, P. Leys, J. Potemans, B. Van den Broeck, E. Van Lil, A. Van de Capelle Advanced Networking Training for Master Students Through OPNET Projects, http://www.esat.kuleuven.be/telemic/networking/opnetwork03_johan.pdf (последно посетен на 18.03.2013)
[47] J. Thomas, "Class C Subnetting Tutorial", http://www.omnisecu.com/tcpip/internet-layer-ip-subnetting-part1.htm (последно посетен на 18.03.2013)
[48] J. Thomas, "Comparison between TCP/IP and OSI", http://www.omnisecu.com/tcpip/tcpip-model.htm (последно посетен на 18.03.2013)
[49] J. Thomas, "Internet Layer - IP Addresses", http://www.omnisecu.com/tcpip/internet-layer-ip-addresses.htm (последно посетен на 18.03.2013)
[50] J. Thomas, "TCP/IP - Internet Layer", http://www.omnisecu.com/tcpip/internet-layer.htm (последно посетен на 18.03.2013)
[51] J. Thomas, "Variable Length Subnet Masking (VLSM)", http://www.omnisecu.com/tcpip/variable-length-subnet-masking-vlsm.html (последно посетен на 18.03.2013)
[52] J.Sklenar, "INTRODUCTION TO OOP IN SIMULA", http://staff.um.edu.mt/jskl1/talk.html (последно посетен на 18.03.2013)
[53] Janitor J., Jakab F., Kniewald K., „Visual Learning Tools for Teaching/Learning Computer Networks: Cisco Networking Academy and Packet Tracer", 2010 Sixth International Conference on Networking and Services, 351-355 p., 7-13 March 2010
[54] John T.Gorgone, „B-ISDN", http://cis.bentley.edu/jgorgone/cs340/C/pdf/BISDN.pdf (последно посетен на 18.03.2013)
[55] Kartik Krishnan, „Computer Networks and Computer Security", http://www4.ncsu.edu/~kksivara/sfwr4c03/lectures/lecture9.pdf (последно посетен на 2.03.2014)

[56] Koohong Kang, Cheeha Kim, "Performance analysis of statistical multiplexing of heterogeneous discrete-time Markovian arrival processes in an ATM network", Computer Communications, Volume 20, Issue 11, 15 October 1997, Pages 970–978
[57] Microsoft, "DHCP Client States in the Lease Process", http://technet.microsoft.com/en-us/library/cc958935.aspx (последно посетен на 18.03.2013)
[58] Microsoft, "Rebinding Time Value (T2)", http://technet.microsoft.com/en-us/library/cc977384.aspx (последно посетен на 18.03.2013)
[59] Microsoft, "Renewal Time Value (T1)", http://technet.microsoft.com/en-us/library/cc959859.aspx (последно посетен на 18.03.2013)
[60] Multiplexing and Demultiplexing, http://comsci.liu.edu/~jrodriguez/cs154fl08/Slides/Lecture5.pdf (последно посетен на 18.03.2013)
[61] Multiplexing, http://www2.cs.uidaho.edu/~krings/CS420/Notes.S10/420-10-06.pdf (последно посетен на 18.03.2013)
[62] NET-SEAL, "Networking Animations", http://www.net-seal.net/animations.php (последно посетен на 18.03.2013)
[63] P. Srisuresh, K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", http://www.rfc-editor.org/rfc/rfc3022.txt (последно посетен на 18.03.2013)
[64] Pablo Brenner, "A Technical tutorial on the IEEE 802.11 Protocol", BreezeCOM, 1997, http://www.sss-mag.com/pdf/802_11tut.pdf (последно посетен на 18.03.2013)
[65] Peter Stark, "T-Carrier", http://www.users.cloud9.net/~stark/ct18.pdf (последно посетен на 18.03.2013)
[66] Public Key Cryptography, http://www.facweb.iitkgp.ernet.in/~sourav/PublicKeyCrypto.pdf (последно посетен на 2.03.2014)
[67] PUBLIC SWITCHED TELEPHONE NETWORK (PSTN), Volume I, http://niits.ru/eng/public/eng-pstn-vol1.pdf (последно посетен на 18.03.2013)
[68] Public Switched Telephone, Network (PSTN), http://www-phare.lip6.fr/~trnguyen/teaching/2010-2011/ptel-puf/PSTN.pdf (последно посетен на 18.03.2013)
[69] Pulse code modulation standards, http://www.irig106.org/docs/106-05/chapter4.pdf (последно посетен на 18.03.2013)
[70] Quantization and Pulse Code Modulation, http://www2.uic.edu/stud_orgs/prof/pesc/part_3_rev_F.pdf (последно посетен на 18.03.2013)
[71] R. Victor Jones, http://people.seas.harvard.edu/~jones/cscie129/nu_lectures/lecture12/T-carriers/T-carriers.html (последно посетен на 18.03.2013)
[72] RadCom, "ATM", http://www.protocols.com/pbook/atm.htm (последно посетен на 18.03.2013)
[73] Rafael Pass, Introduction to Cryptography, http://www.cs.cornell.edu/courses/cs687/2006fa/lectures/lecture1.pdf (последно посетен на 2.03.2014)
[74] Richard M. Roberts, "Networking Fundamentals Course Outline & Text Materials", ISBN: 978-1-59070-449-3, 2005
[75] Rob Pooley, "An Introduction to Programming in Simula", http://www.macs.hw.ac.uk/~rjp/bookhtml/ (последно посетен на 18.03.2013)
[76] Rocky K. C. Chang, „An Integrated View of Teaching and Learning for a Foundational Course on Computer Networking", http://www4.comp.polyu.edu.hk/~csrchang/neted_03_2.pdf (последно посетен на 18.03.2013)

[77] Rocky K. C. Chang, „Teaching Computer Networking with the Help of Personal Computer Networks", http://www4.comp.polyu.edu.hk/~csrchang/personalCN.pdf (последно посетен на 18.03.2013)
[78] Salil Vadhan, Alon Rosen, Private-Key Encryption: Perfect Secrecy, http://people.seas.harvard.edu/~salil/cs127/fall06/docs/lec3.pdf (последно посетен на 2.03.2014)
[79] Smita Rai, Anpeng Huang, Suman Sarkar, "SONET/SDH", http://networks.cs.ucdavis.edu/~mukherje/289i_sq12/SONET.pdf (последно посетен на 18.03.2013)
[80] Stein Krogdahl, „Concepts and terminology in the Simula Programming Language", http://folk.uio.no/simula67/Archive/concepts.pdf (последно посетен на 18.03.2013)
[81] Stephen Kent, Charles Lynn, Joanne Mikkelson, Karen Seo, "Secure Border Gateway Protocol (S-BGP) — Real World Performance and Deployment Issues", http://users.ece.cmu.edu/~adrian/731-sp04/readings/KLMS-SBGP.pdf (последно посетен на 18.03.2013)
[82] Surasak Sanguanpong, "Digital Carrier Systems", http://hccc.ee.ccu.edu.tw/courses/datacom/lecture-vg/carrier.pdf (последно посетен на 18.03.2013)
[83] Surasak Sanguanpong, "Switching Network", http://www.cpe.ku.ac.th/~nguan/presentations/datacom/switch.pdf (последно посетен на 18.03.2013)
[84] Tanenbaum, A., "Computer Networks 3 ed.", Prentice-Hall, 1996
[85] Tektronix, "Synchronous Optical Network (SONET)", http://www.nada.kth.se/kurser/kth/2D1491/02/papper/MO3CB9ABA30C600D1.pdf (последно посетен на 18.03.2013)
[86] The Data Encryption Standard (DES), http://www.facweb.iitkgp.ernet.in/~sourav/DES.pdf (последно посетен на 2.03.2014)
[87] The Joint Task Force on Computing Curricula Association for Computing Machinery IEEE-Computer Society, "Computer Science Curricula 2013", http://ai.stanford.edu/users/sahami/CS2013/strawman-draft/cs2013-strawman.pdf (последно посетен на 18.03.2013)
[88] Time Division Multiplexing (TDM), http://faraday.ee.emu.edu.tr/eaince/ee360/lecture_notes/LECTURE_NOTES_11.pdf (последно посетен на 18.03.2013)
[89] Tommy Svensson, Alex Popescu, „Development of laboratory exercises based on OPNET Modeler", http://staff.ustc.edu.cn/~bhua/experiments/Lab_Exercices_Modeler.pdf (последно посетен на 18.03.2013)
[90] VIVACOM, "Данни и Интернет", http://www.vivacom.bg/bg/business/prices_and_services/danni_i_internet (последно посетен на 18.03.2013)
[91] Wikipedia, "OSI model", http://en.wikipedia.org/wiki/OSI_model (последно посетен на 18.03.2013)
[92] Wireless Networking Basics, http://documentation.netgear.com/reference/fra/wireless/pdfs/Chapter.pdf (последно посетен на 2.03.2014)
[93] Дебра Шиндър, „Компютърни мрежи", София, 2009 г.
[94] Иван Ганчев, 'Компютърни мрежи и комуникации', Пловдив, 1999 г.
[95] Мерджанов П., Телекомуникационни мрежи, „Нови Знания", София, 2002.
[96] Мирчев С., АТМ комутации, „Нови Знания", София, 2001.

[97] Питър Нортън, „Пълно ръководство за работа с мрежи”, София, 1999 г.
[98] Пулков Вл., PDH и SDH – цифрови йерархии в телекомуникациите, „Нови Знания“, София, 1998.
[99] Стоянова В., Преносни линии и мрежи, „Техника“, 1994
[100] Цанков Б., Телекомуникации-фиксирани, мобилни и IP, „Нови Знания“, София, 2006.