

Г.З. Халімов

КОНСПЕКТ ЛЕКЦІЙ з дисципліни ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ СИСТЕМИ (частина I)



Харків 2019

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

КОНСПЕКТ ЛЕКЦІЙ
з дисципліни
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ СИСТЕМИ
(частина І)

для студентів усіх форм навчання спеціальності 125 «Кібербезпека»
за освітньою програмою «Безпека інформаційних комунікаційних систем»

Електронне видання

ЗАТВЕРДЖЕНО
кафедрою БІТ.
Протокол № 12 від 04.04.2019

Харків 2019

Конспект лекцій з дисципліни «Інформаційно-комунікаційні системи», частина І, для студентів усіх форм навчання спеціальності 125 «Кібербезпека» за освітньою програмою «Безпека інформаційних комунікаційних систем» [Електронний ресурс] / Упоряд. Г.З. Халімов. – Електронне видання. – Харків: ХНУРЕ, 2019. – 207 с. – pdf 7,36 Мб.

Упорядник Г.З. Халімов

Рецензент О.В. Сєверінов, канд. техн. наук, доцент, доцент каф. БІТ

ЗМІСТ

Передмова	5
Лекція 1. Загальні відомості.....	6
1.1 Проблеми побудови комп'ютерних мереж	6
1.2 Топологія комп'ютерних мереж	14
1.3 Логічна та фізична структуризація.....	31
1.4 Контрольні запитання та завдання	33
Лекція 2. Архітектура комп'ютерних систем.....	35
2.1 Модель взаємодії відкритих систем	35
2.2 Функції рівнів моделі OSI	44
2.3 Контрольні запитання та завдання	55
Лекція 3. Стандартні стеки комунікаційних протоколів.....	57
3.1 Стандартні стеки комунікаційних протоколів	57
3.2 Вимоги, які висуваються до обчислювальних мереж	62
3.3 Контрольні запитання та завдання	73
Лекція 4. Характеристика фізичного рівня архітектури комп'ютерних мереж	74
4.1 Типи ліній зв'язку	74
4.2 Характеристики ліній зв'язку	79
4.3 Контрольні запитання та завдання	92
Лекція 5. Методи передачі дискретних даних на фізичному рівні	93
5.1 Стандарти кабелів зв'язку	93
5.2 Методи аналогової модуляції	102
5.3 Контрольні запитання та завдання	108
Лекція 6. Методи цифрового кодування.....	109
6.1 Вимоги до методів цифрового кодування	109
6.2 Цифрові коди	110
6.3 Контрольні запитання та завдання	126
Лекція 7. Методи кадрування	127
7.1 Знакове кадрування.....	127
7.2 Бітове кадрування.....	131

7.3 Вибір розміру кадра	134
7.4 Контрольні запитання та завдання	138
Лекція 8. Методи комутації.....	139
8.1 Класифікація методів комутації	139
8.2 Комутація каналів	141
8.3 Комутація пакетів.....	149
8.4 Контрольні запитання та завдання	152
Лекція 9. Методи комутації пакетів	153
9.1 Комутація пакетів з використанням методу віртуальної передачі	153
9.2 Комутація повідомлень.....	157
9.3 Передача із встановленням та без встановлення з'єднань.....	159
9.4 Контрольні запитання та завдання	161
Лекція 10. Методи повторної передачі даних	162
10.1 Методи забезпечення достовірної передачі пакетів у мережі	162
10.2 Метод передачі з зупинкою та очікуванням	165
10.3 Контрольні запитання та завдання	168
Лекція 11. Принципи побудови мережного рівня.....	169
11.1 Побудова мережі на основі протоколів мережного рівня.....	169
11.2 Принципи маршрутизації	174
11.3 Контрольні запитання та завдання	179
Лекція 12. Протоколи маршрутизації.....	180
12.1 Класифікація алгоритмів побудови таблиць маршрутизації	180
12.2 Функції маршрутизатора	185
12.3 Реалізація міжмережної взаємодії засобами TCP/IP	189
12.4 Контрольні запитання та завдання	190
Лекція 13. Принципи побудови стеку TCP/IP	192
13.1 Функції стеку TCP/IP	192
13.2 Організація доменів та доменних імен	198
13.3 Система доменних імен DNS	202
13.4. Контрольні запитання та завдання	205
Рекомендована література.....	206

ПЕРЕДМОВА

Електронний конспект лекцій з дисципліни «Інформаційно-комунікаційні системи» містить основні відомості щодо архітектури комп'ютерних мереж, технологій побудови локальних мереж, мережних протоколів, математичних моделей мережного обслуговування, методів забезпечення якості обслуговування, маршрутизації, побудови мережних топологій.

Курс лекцій розділений на дві частини за семестрами навчання. В цілому обмежений вибір тем лекцій визначається навчальною програмою підготовки. Наприкінці викладення лекцій надано ряд запитань для перевірки знань студентів. Практичні вправи винесені на практичні та лабораторні заняття.

Зміст електронного конспекту на матеріалі двох фундаментальних підручників: «Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов» (В.Г. Олифер, Н.А. Олифер) та «Сети передачи данных» (Д. Бертсекас, Р. Галагер).

Основні концепції та принципи розглянуто за допомогою простих математичних засобів, моделей і аналізу. Для ілюстрації ідей широко використовуються рисунки, приклади, що пояснюють більш тонкі деталі. Включення математичного аналізу забезпечує глибше розуміння матеріалу і розвиток здібностей до досліджень у цій галузі.

Частина курсу лекцій присвячена викладенню питань забезпечення якості обслуговування в мережах з комутацією пакетів, маршрутизації, побудови мережних топологій із заданими властивостями. Математичний аналіз побудований на основі методів множинного доступу, моделей графів і масового обслуговування. Вивчення алгоритмів маршрутизації ґрунтується на оптимізаційних методах пошуку екстремумів, синтезі топологій на базі лінійного програмування.

Попри обмеженість лекційних тем, їх вибір достатній для формування базових знань у цій галузі та є демонстрацією класичної роботи математичних методів під час пошуку мережних рішень.

Лекція 1

ЗАГАЛЬНІ ВІДОМОСТІ

1.1 Проблеми побудови комп'ютерних мереж

Під час створення обчислювальних мереж їхнім розробникам довелося вирішити багато проблем. У цьому розділі ми розглянемо тільки найважливіші з них, причому в тій послідовності, у якій вони природно виникали у процесі розвитку й удосконалювання мережних технологій.

Механізми взаємодії комп'ютерів у мережі багато чого запозичили у схеми взаємодії комп'ютера з периферійними пристроями, тому почнемо розгляд принципів роботи мережі з цього «домережного» випадку.

Зв'язок комп'ютера з периферійними пристроями

Для обміну даними між комп'ютером і периферійним пристроєм (ПП) у комп'ютері передбачено зовнішній *інтерфейс* (рис. 1.1), тобто набір проводів, що з'єднують комп'ютер і периферійний пристрій, а також набір правил обміну інформацією по цих проводах (іноді замість терміна *інтерфейс* застосовується термін *протокол* – детальніше про ці важливі терміни ми ще поговоримо). Прикладами інтерфейсів, які використовуються у комп'ютерах, є паралельний інтерфейс Centronics, призначений, як правило, для підключення принтерів, і послідовний інтерфейс RS-232C, через який підключаються мишка, модем та інші пристрої. З боку комп'ютера інтерфейс реалізується сукупністю апаратних та програмних засобів: контролером ПП і спеціальною програмою, що управляє цим контролером, які часто називають *драйвером* відповідного периферійного пристрою.

З боку ПП інтерфейс найчастіше реалізується апаратним пристроєм управління, хоча зустрічаються й програмно-керовані периферійні пристрої.

Програма, що виконується процесором, може обмінюватися даними за допомогою команд введення/виведення з будь-якими модулями, підключеними до внутрішньої шини комп'ютера, у тому числі й з контролерами ПП.

Периферійні пристрої можуть приймати від комп'ютера як дані, наприклад, байти інформації, яку потрібно роздрукувати на папері, так і команди управління, у відповідь на які ПП може виконати спеціальні дії, такі як: перевести головку диска на необхідну доріжку або ж виштовхнути аркуш паперу з принтера. Периферійний пристрій використовує зовнішній інтерфейс

комп'ютера не тільки для прийому інформації, але й для передачі інформації в комп'ютер, тобто обмін даними зовнішнім інтерфейсом, як правило, є двонаправленим. Так, наприклад, принтер, який за своєю природою є пристроєм виведення інформації, повертає у комп'ютер дані про свій стан.

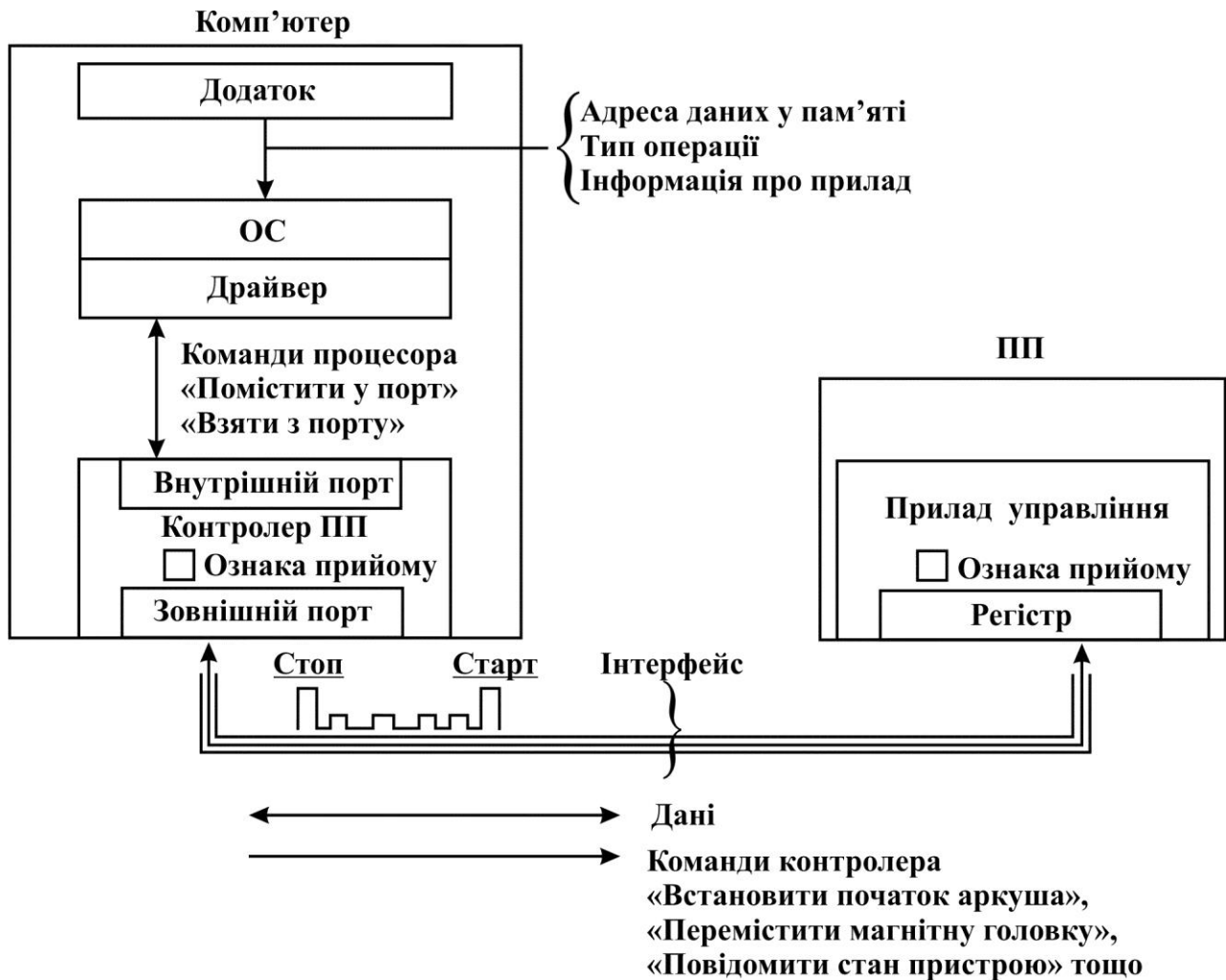


Рисунок 1.1 – Зв'язок комп'ютера з периферійним пристроєм

Контролери ПП приймають команди й дані від процесора у свій внутрішній буфер, який часто називається регістром або портом, потім виконують необхідні перетворення цих даних і команд відповідно до форматів, зрозумілими ПП, і видають їх на зовнішній інтерфейс.

Розподіл обов'язків між контролером і драйвером ПП може бути різним, але зазвичай контролер виконує набір простих команд з управління ПП, а драйвер використовує ці команди, щоб змусити пристрій робити більш складні дії за деяким алгоритмом. Наприклад, контролер принтера може підтримувати такі елементарні команди, як «Друкування символу», «Переведення рядка», «Повернення каретки» тощо. Драйвер же принтера за

допомогою цих команд організує друкування рядків символів, розподіл документа на сторінки та інші більш високорівневі операції. Для того самого контролера можна розробити різні драйвери, які управлятимуть даним ПП не однаково – одні краще, інші гірше – залежно від досвіду та здібностей програмістів, що їх розробляли.

Розглянемо схему передачі одного байта інформації від прикладної програми на периферійний пристрій. Програма, якій треба було виконати обмін даними з ПП, звертається до драйвера цього пристрою, повідомляючи йому як параметр адресу байта пам'яті, який потрібно передати. Драйвер завантажує значення цього байта в буфер контролера ПП, що починає послідовно передавати біти в лінію зв'язку, подаючи кожний біт відповідним електричним сигналом. Щоб пристрою управління ПП стало зрозуміло, що починається передача байта, перед передачею першого біта інформації контролер ПП формує стартовий сигнал специфічної форми, а після передачі останнього інформаційного біта – столовий сигнал. Ці сигнали *синхронізують* передачу байта.

Крім інформаційних біт, контролер може передавати біт контролю парності для підвищення вірогідності обміну. Пристрій управління, виявивши на відповідній лінії стартовий біт, виконує підготовчі дії та починає приймати інформаційні біти, формуючи з них байт у своєму прийомному буфері. Якщо передача супроводжується бітом парності, то виконується перевірка правильності передачі: під час правильно виконаної передачі у відповідному регістрі пристрою управління встановлюється ознака завершення прийому інформації.

Зазвичай на драйвер покладають найбільш складні функції протоколу (наприклад, підрахунок контрольної суми послідовності переданих байтів, аналіз стану периферійного пристрою, перевірка правильності виконання команди). Але навіть найпримітивніший драйвер контролера має підтримувати як мінімум дві операції: «Взяти дані з контролера в оперативну пам'ять» і «Передати дані з оперативної пам'яті в контролер».

Існують як досить спеціалізовані інтерфейси, придатні для підключення вузького класу пристроїв (наприклад, графічних моніторів високого дозволу фірми Vista), так і інтерфейси загального призначення, які є стандартними й дозволяють підключати різні периферійні пристрої. Прикладом такого інтерфейсу є інтерфейс RS-232C, що підтримується багатьма терміналами, принтерами, графобудівниками, маніпуляторами типу «мишка» і багатьма іншими пристроями.

Найпростіший випадок взаємодії двох комп'ютерів

У найпростішому випадку взаємодія комп'ютерів може бути реалізована за допомогою тих самих засобів, що використовуються для взаємодії комп'ютера з периферією, наприклад, через послідовний інтерфейс RS-232C. На відміну від взаємодії комп'ютера з периферійним пристроєм, коли програма працює, як правило, тільки з одного боку – з боку комп'ютера, у цьому випадку відбувається взаємодія двох програм, що працюють на кожному з комп'ютерів.

Програма, що працює на одному комп'ютері, не може отримати безпосередній доступ до ресурсів іншого комп'ютера – його дискам, файлам, принтеру. Вона може тільки «попросити» про це програму, що працює на тому комп'ютері, якому належать ці ресурси. Ці «прохання» виражаються у вигляді *повідомлень*, переданих по каналах зв'язку між комп'ютерами. Повідомлення можуть містити не тільки команди на виконання деяких дій, але й власне інформаційні дані (наприклад, вміст деякого файла).

Розглянемо випадок, коли користувачу, що працює з текстовим редактором на персональному комп'ютері А, потрібно прочитати частину деякого файла, розташованого на диску персонального комп'ютера В (рис. 1.2). Припустимо, що ми зв'язали ці комп'ютери кабелем зв'язку через COM-порти, які, як відомо, реалізують інтерфейс RS-232C (таке з'єднання часто називають нуль-модемним). Нехай для визначеності комп'ютери працюють під управлінням MS-DOS, хоча в даному випадку це не принципово.

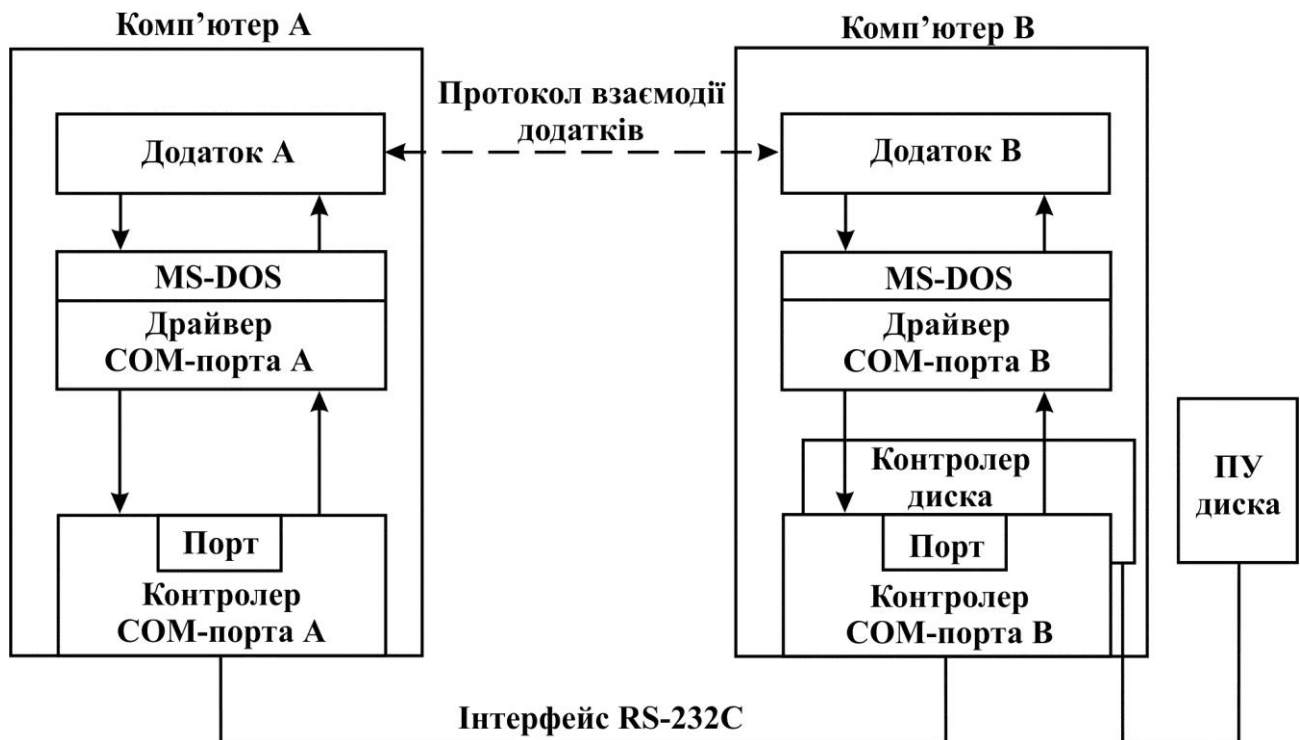


Рисунок 1.2 – Взаємодія двох комп'ютерів

Драйвер СОМ-порта разом з контролером СОМ-порта працюють приблизно так само, як і в описаному вище випадку взаємодії ПП з комп'ютером. Однак при цьому роль пристрою управління ПП виконують контролер і драйвер СОМ-порта іншого комп'ютера. Разом вони забезпечують між комп'ютерами передачу по кабелю одного байта інформації. (У «справжніх» локальних мережах подібні функції передачі даних у лінію зв'язку виконуються мережними адаптерами та їхніми драйверами.)

Драйвер комп'ютера В періодично опитує ознаку завершення прийому, яка встановлюється контролером при правильно виконаній передачі даних, і, як тільки вона з'явиться, зчитує прийнятий байт із буфера контролера в оперативну пам'ять, тим самим роблячи її доступною для програм комп'ютера В. У деяких випадках драйвер викликається асинхронно, за перериваннями від контролера.

Отже, у розпорядженні програм комп'ютерів А і В є засіб для передачі одного байта інформації. Але розглянута у нашому прикладі задача значно складніша, оскільки потрібно передати не один байт, а певну частину заданого файла. Усі пов'язані із цим додаткові проблеми мають вирішитися програмами більш високого рівня, ніж драйвери СОМ-портів. Для визначеності назовемо такі програми комп'ютерів А і В додатком А та додатком В відповідно. Отже, додаток А має сформулювати повідомлення-запит для додатка В. У запиті необхідно вказати ім'я файлу, тип операції (у нашому випадку – читання), зсув та розмір області файлу, яка містить потрібні дані.

Щоб передати повідомлення комп'ютеру В, додаток А звертається до драйвера СОМ-порта, повідомляючи йому адресу в оперативній пам'яті, за якою драйвер знаходить повідомлення й потім передає його байт за байтом додатку В. Додаток В, прийнявши запит, виконує його, зчитує необхідну область файлу з диска за допомогою засобів локальної ОС у буферну область своєї оперативної пам'яті, а далі за допомогою драйвера СОМ-порта передає зчитані дані каналом зв'язку в комп'ютер А, де вони й попадають до додатка А.

Описані функції додатка А могла б виконати сама програма текстового редактора, але включати ці функції до складу кожного додатка – текстових редакторів, графічних редакторів, систем управління базами даних та інших додатків, яким потрібен доступ до файлів, – не раціонально (хоча існує велика кількість програм, які дійсно самостійно вирішують всі задачі міжмашинного обміну даними, наприклад Kermit – програма обміну файлами через СОМ-порти, реалізована для різних ОС, Norton Commander 3.0 з його функцією Link). Вигідніше створити спеціальний програмний модуль, який виконуватиме функції формування повідомлень-запитів і прийому результатів для всіх

додатків комп'ютера. Такий службовий модуль називається клієнтом. З боку комп'ютера В має працювати інший модуль – сервер, який постійно очікує приходу запитів на віддалений доступ до файлів, розташованих на диску цього комп'ютера. Сервер, прийнявши запит з мережі, звертається до локального файла й виконує з ним задані дії, можливо, за участю локальної ОС.

Програмні клієнт і сервер виконують системні функції з обслуговування запитів додатків комп'ютера А на віддалений доступ до файлів комп'ютера В. Щоб додатки комп'ютера В могли користуватися файлами комп'ютера А, описану схему слід симетрично доповнити клієнтом для комп'ютера В і сервером для комп'ютера А.

Схема взаємодії клієнта й сервера з додатками та операційною системою наведена на рис. 1.3. Незважаючи на те, що ми розглянули дуже просту схему апаратного зв'язку комп'ютерів, функції програм, які забезпечують доступ до віддалених файлів, дуже схожі на функції модулів мережної операційної системи, що працює в мережі з більш складними апаратними зв'язками комп'ютерів.

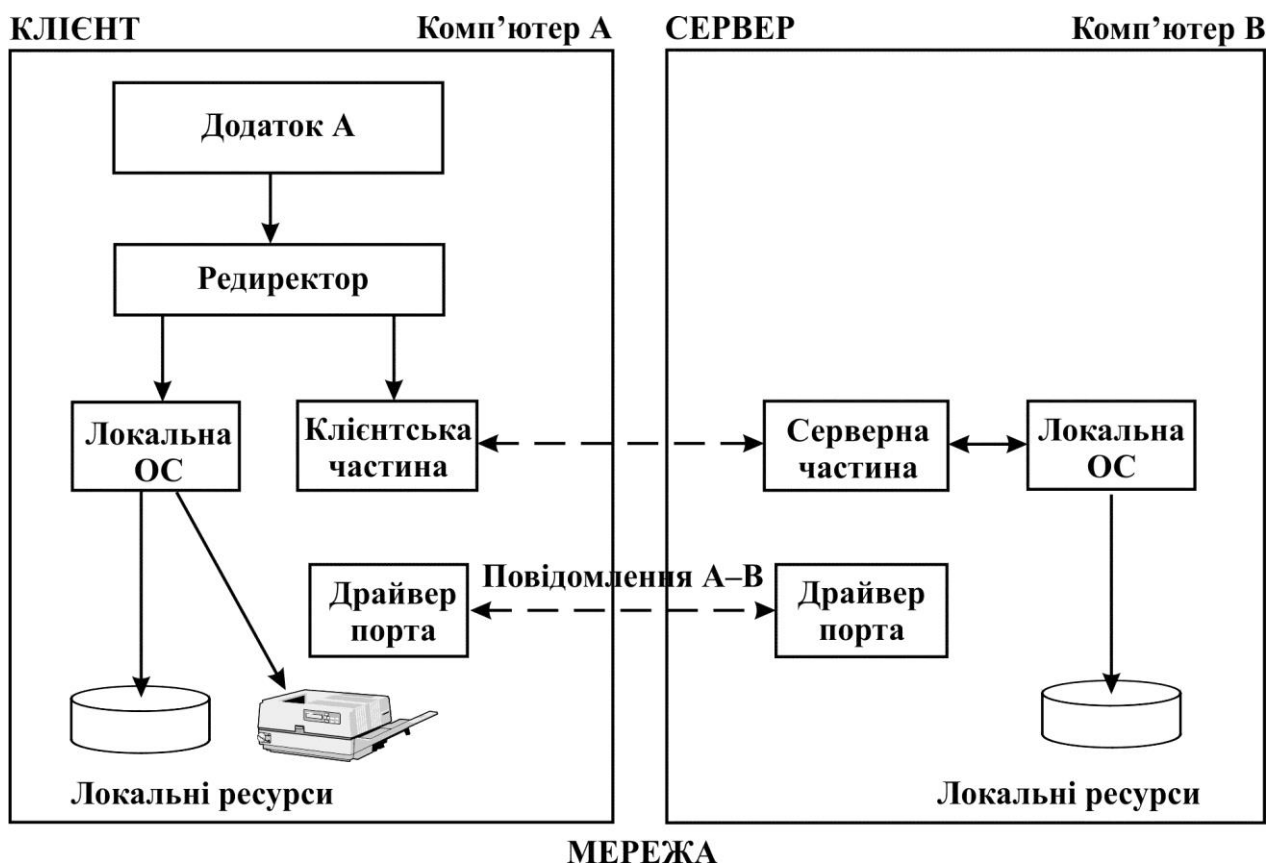


Рисунок 1.3 – Взаємодія програмних компонентів при зв'язку двох комп'ютерів

Дуже зручною й корисною функцією клієнтської програми є здатність відрізнити запит до віддаленого файлу від запиту до локального файлу. Якщо клієнтська програма вміє це робити, то додатки не мають піклуватися про те, з яким файлом вони працюють (локальним або віддаленим), клієнтська програма сама розпізнає й *перенаправляє (redirect)* запит до віддаленої машини. Звідси й назва, яка часто використовується для клієнтської частини мережної ОС, – *редиректор*. Іноді функції розпізнавання виділяються в окремий програмний модуль, у цьому випадку редиректором називають не всю клієнтську частину, а тільки цей модуль.

Проблеми фізичної передачі даних по лініях зв'язку

Навіть під час розгляду найпростішої мережі, що складається всього з двох машин, можна побачити багато проблем, властивих для будь-якої обчислювальної мережі, у тому числі проблеми, пов'язані з фізичною передачею сигналів по лініях зв'язку, без вирішення якої унеможлиблюється будь-який вид зв'язку.

В обчислювальній техніці для подання даних використовується двійковий код. У комп'ютері одиницям і нулям даних відповідають дискретні електричні сигнали. Подання даних у вигляді електричних або оптичних сигналів називається *кодуванням*. Існують різні способи кодування двійкових цифр 1 і 0, наприклад, потенційний спосіб, під час якого одиниці відповідає один рівень напруги, а нулю – інший, або імпульсний спосіб, коли для подання цифр використовуються імпульси однієї або різної полярності.

Аналогічні підходи можуть бути використані для кодування даних і передачі їх між двома комп'ютерами по лініях зв'язку. Однак ці лінії зв'язку відрізняються за своїми електричними характеристиками від тих, що існують усередині комп'ютера. Головна відмінність зовнішніх ліній зв'язку від внутрішніх є в їхній набагато більшій довжині, а також у тому, що вони проходять поза екранованим корпусом по просторах, які найчастіше піддаються впливу сильних електромагнітних завад. Все це призводить до значних перекручувань прямокутних імпульсів (наприклад, «завалюванню» фронтів), ніж усередині комп'ютера. Тому для надійного розпізнавання імпульсів на прийомному кінці лінії зв'язку в ході передачі даних усередині й поза комп'ютером не завжди можна використовувати одні й ті самі швидкості та способи кодування. Наприклад, повільне наростання фронту імпульса через високе ємнісне навантаження лінії вимагає передачі імпульсів з меншою

швидкістю (щоб передній і задній фронти сусідніх імпульсів не перекривалися й імпульс встиг дорости до необхідного рівня).

В обчислювальних мережах застосовують як потенційне, так і імпульсне кодування дискретних даних, а також специфічний спосіб подання даних, який ніколи не використовується усередині комп'ютера, – *модуляцію* (рис. 1.4). Під час модуляції дискретна інформація подається синусоїдальним сигналом тієї частоти, яку добре передає наявна лінія зв'язку.

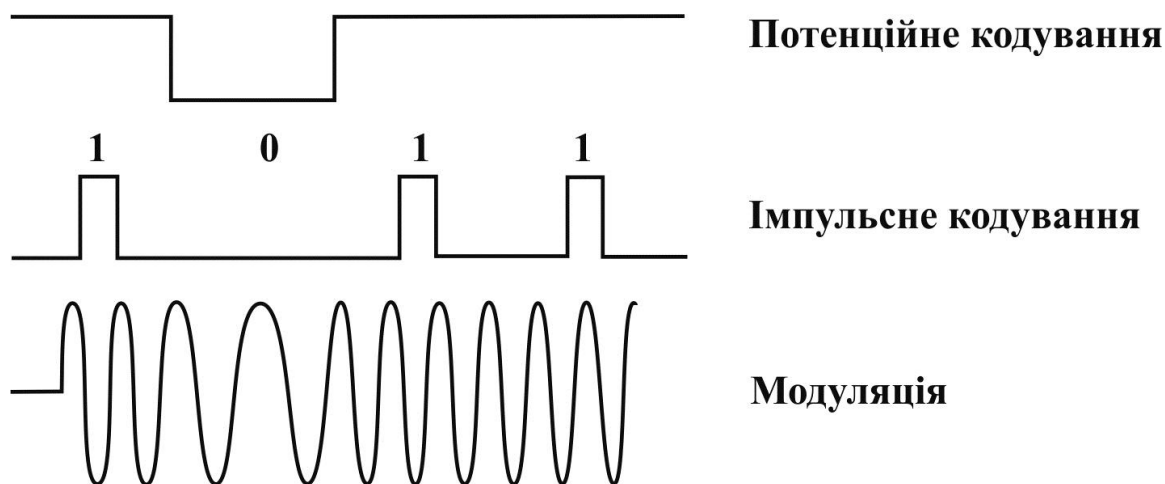


Рисунок 1.4 – Приклади подання дискретної інформації

Потенційне або імпульсне кодування застосовується на каналах високої якості, а модуляція на основі синусоїдальних сигналів переважно в тому випадку, коли канал вносить сильні перекручування в передані сигнали. Зазвичай модуляція використовується в глобальних мережах під час передачі даних через аналогові телефонні канали зв'язку, які були розроблені для передачі голосу в аналоговій формі й тому погано підходять для безпосередньої передачі імпульсів.

На спосіб передачі сигналів впливає й кількість проводів у лініях зв'язку між комп'ютерами. Для скорочення вартості ліній зв'язку в мережах зазвичай прагнуть до скорочення кількості проводів і через це використовують не паралельну передачу всіх біт одного байта або навіть декількох байт, як це робиться усередині комп'ютера, а послідовну, побітну передачу, що вимагає лише одну пару проводів.

Ще однією проблемою, яку потрібно вирішувати в ході передачі сигналів, є проблема взаємної *синхронізації* передавача одного комп'ютера з приймачем іншого. В ході організації взаємодії модулів усередині комп'ютера ця проблема вирішується дуже просто, оскільки в цьому випадку всі модулі синхронізуються від загального тактового генератора. Проблема синхронізації під час зв'язку

комп'ютерів може вирішуватися різними способами, як за допомогою обміну спеціальними тактовими синхроімпульсами окремою лінією, так і за допомогою періодичної синхронізації заздалегідь обумовленими кодами або імпульсами характерної форми, яка відрізняється від форми імпульсів даних.

Незважаючи на заходи, що вживаються, – вибір відповідної швидкості обміну даними, ліній зв'язку з певними характеристиками, способу синхронізації приймача й передавача, – існує імовірність перекручування деяких біт переданих даних. Для підвищення надійності передачі даних між комп'ютерами часто використовується стандартний прийом – підрахунок *контрольної суми* й передача її по лініях зв'язку після кожного байта або після деякого блока байтів. Часто до протоколу обміну даними включається як обов'язковий елемент сигнал-квитанція, що підтверджує правильність прийому даних і посилає від одержувача відправнику.

Задачі надійного обміну двійковими сигналами, поданими відповідними електромагнітними сигналами, в обчислювальних мережах вирішує певний клас обладнання. У локальних мережах це *мережні адаптери*, а в глобальних – апаратура передачі даних, до якої належать, наприклад, пристрої, що виконують модуляцію й демодуляцію дискретних сигналів, – *модеми*. Це обладнання кодує й декодує кожен інформаційний біт, синхронізує передачу електромагнітних сигналів по лініях зв'язку, перевіряє правильність передачі по контрольній сумі й може виконувати деякі інші операції. Мережні адаптери розраховані, як правило, на роботу з певним *передавальним середовищем* – коаксіальним кабелем, крученою парою, оптоволоком тощо. Кожний тип передавального середовища має певні електричні характеристики, які впливають на спосіб використання даного середовища, і визначає швидкість передачі сигналів, спосіб їхнього кодування та деякі інші параметри.

1.2 Топологія комп'ютерних мереж

При об'єднанні в мережу більшої кількості комп'ютерів виникає цілий комплекс нових проблем.

Топологія фізичних зв'язків

У першу чергу необхідно обрати спосіб організації фізичних зв'язків, тобто *топологію*. Під топологією обчислювальної мережі розуміють конфігурацію графа, вершинам якого відповідають комп'ютери мережі (іноді й інше обладнання, наприклад концентратори), а ребрам – фізичні зв'язки між

ними. Комп'ютери, підключені до мережі, часто називають *станціями* або *вузлами мережі*.

Зазначимо, що конфігурація *фізичних зв'язків* визначається електричними з'єднаннями комп'ютерів між собою і може відрізнятися від конфігурації *логічних зв'язків* між вузлами мережі. Логічні зв'язки є маршрутами передачі даних між вузлами мережі й утворюються шляхом відповідного налаштування комунікаційного обладнання.

Вибір топології електричних зв'язків істотно впливає на характеристики мережі. Наприклад, наявність резервних зв'язків підвищує надійність мережі та дає можливість балансувати завантаження окремих каналів. Простота приєднання нових вузлів, властива деяким топологіям, робить мережу легко розширюваною. Економічні розмірковування часто призводять до вибору топологій, для яких характерна мінімальна сумарна довжина ліній зв'язку.

Розглянемо ті топології, що зустрічаються найчастіше.

Повнозв'язна топологія (рис. 1.5, а) відповідає мережі, де кожен комп'ютер мережі зв'язаний з усіма іншими. Незважаючи на логічну простоту, цей варіант виявляється громіздким та неефективним. Дійсно, кожен комп'ютер у мережі повинен мати велику кількість комунікаційних портів, достатню для зв'язку з кожним комп'ютером мережі. Для кожної пари комп'ютерів має бути виділена окрема електрична лінія зв'язку. Повнозв'язні топології застосовуються рідко, оскільки не відповідають жодній з наведених вище вимог. Частіше цей вид топології використовується в багатомашинних комплексах або глобальних мережах за невеликої кількості комп'ютерів.

Всі інші варіанти базуються на неповнозв'язних топологіях, коли для обміну даними між двома комп'ютерами може знадобитися проміжна передача даних через інші вузли мережі.

Коміркова топологія (*mesh*) виходить із повнозв'язної шляхом вилучення деяких можливих зв'язків (рис. 1.5, б). У мережі з комірковою топологією безпосередньо зв'язуються тільки ті комп'ютери, між якими відбувається інтенсивний обмін даними, а для обміну даними між комп'ютерами, не з'єднаними прямими зв'язками, використовуються транзитні передачі через проміжні вузли. Коміркова топологія допускає з'єднання великої кількості комп'ютерів і характерна, як правило, для глобальних мереж.

Спільна шина (рис. 1.5, е) є дуже розповсюдженою (а донедавна найпоширенішою) топологією для локальних мереж. У цьому випадку комп'ютери підключаються до одного коаксіального кабелю за схемою «монтажного АБО». Інформація, що передається, може поширюватися в обидва

боки. Застосування спільної шини знижує вартість проводки, уніфікує підключення різних модулів, забезпечує можливість майже миттєвого широкомовного звернення до всіх станцій мережі. Отже, основними перевагами такої схеми є дешевизна й простота розведення кабелю по приміщеннях. Найсерйозніший недолік спільної шини – її низька надійність: будь-який дефект кабелю або якогось із численних рознімів повністю паралізує всю мережу. На жаль, дефект коаксіального рознімання не рідкість. Ще одним недоліком спільної шини є її невисока продуктивність, оскільки при такому способі підключення в кожний момент часу тільки один комп'ютер може передавати дані в мережу. Тому пропускна здатність каналу зв'язку завжди ділиться тут між всіма вузлами мережі.

Топологія *зірка* (рис. 1.5, г). У цьому випадку кожний комп'ютер підключається окремим кабелем до загального пристрою, який називається *концентратором* і знаходиться в центрі мережі. У функції концентратора входить направлення переданої комп'ютером інформації одному або всім іншим комп'ютерам мережі. Головна перевага цієї топології перед спільною шиною – це значно більша надійність. Будь-які неприємності з кабелем стосуються лише того комп'ютера, до якого цей кабель приєднаний, і тільки несправність концентратора може вивести з ладу всю мережу. Крім того, концентратор може відігравати роль інтелектуального фільтра інформації, яка надходить від вузлів у мережу, і при необхідності блокувати заборонені адміністратором передачі.

До недоліків топології типу *зірка* належить більш висока вартість мережного обладнання через необхідність придбання концентратора. Крім того, можливості з нарощування кількості вузлів у мережі обмежуються кількістю портів концентратора. Іноді має сенс будувати мережу з використанням декількох концентраторів, ієрархічно з'єднаних між собою зв'язками типу «зірка» (рис. 1.5, д). У цей час ієрархічна зірка є найпоширенішим типом топології зв'язків, як у локальних, так і глобальних мережах.

У мережах з *кільцевою* конфігурацією (рис. 1.5, е) дані передаються по кільцю від одного комп'ютера до іншого, як правило, в одному напрямі. Якщо комп'ютер розпізнає дані як «свої», то він копіює їх собі у внутрішній буфер. У мережі з кільцевою топологією необхідно вживати спеціальних заходів, щоб у випадку виходу з ладу або відключення будь-якої станції не перервався канал зв'язку між іншими станціями. Кільце є дуже зручною конфігурацією для організації зворотного зв'язку – дані, зробивши повний оберт, повертаються до вузла-джерела. Тому цей вузол може контролювати процес доставки даних

адресату. Часто ця властивість кільця використовується для тестування зв'язності мережі та пошуку вузла, який працює некоректно. Для цього в мережу посилають спеціальні тестові повідомлення.

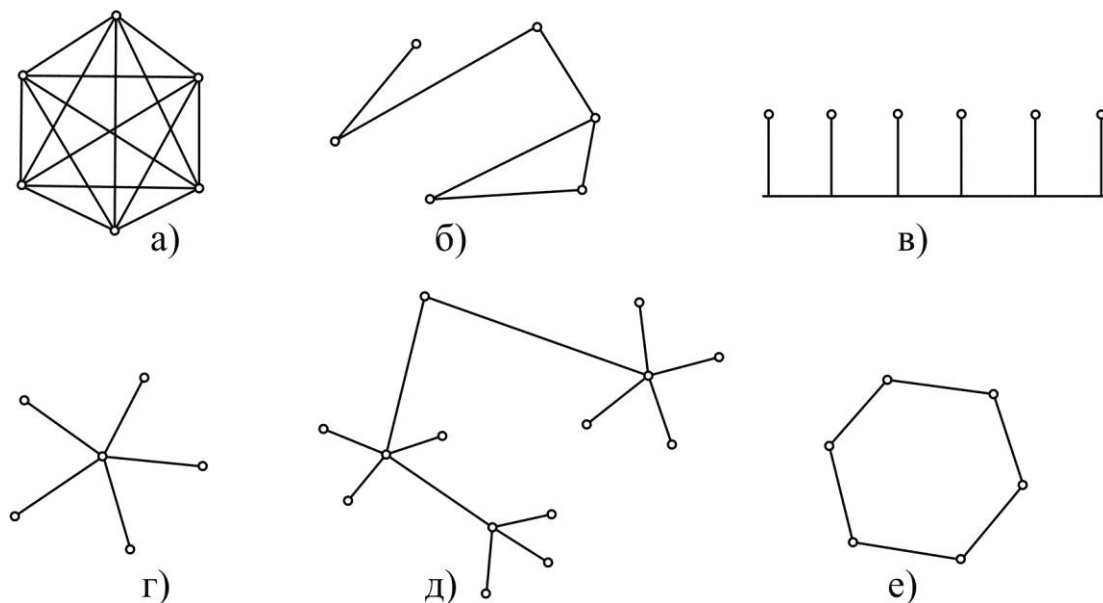


Рисунок 1.5 – Типові топології мереж

У той час, як невеликі мережі, як правило, мають типову топологію – зірка, кільце або спільна шина, для великих мереж характерна наявність довільних зв'язків між комп'ютерами. У таких мережах можна виділити окремі довільно зв'язані фрагменти (підмережі), які мають типову топологію, тому їх називають мережами зі *змішаною топологією* (рис. 1.6).

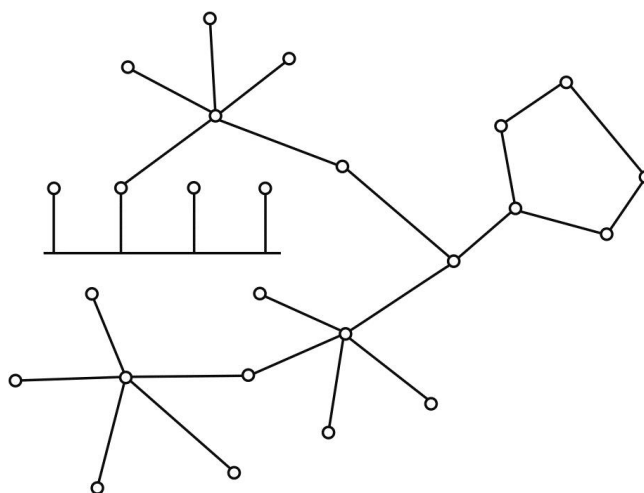


Рисунок 1.6 – Змішана топологія

Організація спільного використання ліній зв'язку

Тільки в мережі з повнозв'язаною топологією для з'єднання кожної пари комп'ютерів є окрема лінія зв'язку. В усіх інших випадках неминуче виникає

питання, як організувати спільне використання ліній зв'язку декількома комп'ютерами мережі. Як і завжди в ході поділу ресурсів головною метою тут є здешевлення мережі.

В обчислювальних мережах використовують як індивідуальні лінії зв'язку між комп'ютерами, так і *поділювані (shared)*, коли одна лінія зв'язку поперемінно використовується декількома комп'ютерами. У випадку застосування поділюваних ліній зв'язку (часто використовується також термін «поділюване середовище передачі даних» – *shared media*) виникає комплекс проблем, пов'язаних з їхнім спільним використанням, що включає як суто електричні проблеми забезпечення необхідної якості сигналів при підключенні до того самого проведення декількох приймачів і передавачів, так і логічні проблеми поділу в часі доступу до цих ліній.

Класичним прикладом мережі з поділюваними лініями зв'язку є мережі з топологією «спільна шина», у яких один кабель спільно використовується всіма комп'ютерами мережі. Жоден з комп'ютерів мережі в принципі не може індивідуально, незалежно від всіх інших комп'ютерів мережі, використовувати кабель, оскільки при одночасній передачі даних відразу декількома вузлами сигнали змішуються й перекручуються. У топологіях «кільце» або «зірка» індивідуальне використання ліній зв'язку, які з'єднують комп'ютери, принципово можливе, але ці кабелі часто також розглядають як поділювані для всіх комп'ютерів мережі так, що, наприклад, тільки один комп'ютер кільця має право в цей момент часу відправляти по кільцю пакети іншим комп'ютерам.

Існують різні способи вирішення задачі організації спільного доступу до поділюваних ліній зв'язку. Всередині комп'ютера проблеми поділу ліній зв'язку між різними модулями також існують, – прикладом є доступ до системної шини, яким управляє або процесор, або спеціальний арбітр шини. У мережах організація спільного доступу до ліній зв'язку має свою специфіку через істотно більший час поширення сигналів довгими проводами, до того ж цей час для різних пар комп'ютерів може бути різним. Через це процедури узгодження доступу до лінії зв'язку можуть займати занадто великий проміжок часу й призводити до значних втрат продуктивності мережі.

Незважаючи на всі ці складності, у локальних мережах поділювані лінії зв'язку використовуються дуже часто. Цей підхід, зокрема, реалізований у широко розповсюджених класичних технологіях Ethernet і Token Ring. Однак в останні роки намітилася тенденція відмови від поділюваних середовищ передачі даних і в локальних мережах. Це пов'язано з тим, що за здешевлення, яке досягається таким чином, мережі доводиться розплачуватися продуктивністю.

Мережа з поділюваним середовищем при великій кількості вузлів працюватиме завжди повільніше, ніж аналогічна мережа з індивідуальними лініями зв'язку, оскільки пропускна здатність індивідуальної лінії зв'язку дістається одному комп'ютеру, а в ході її спільного використання – ділиться на всі комп'ютери мережі. Часто з такою втратою продуктивності миряться заради збільшення економічної ефективності мережі. Не тільки в класичних, але й у зовсім нових технологіях, розроблених для локальних мереж, зберігається режим поділюваних ліній зв'язку. Наприклад, розробники технології Gigabit Ethernet, прийнятої в 1998 році як новий стандарт, включили режим поділу передавального середовища у свої специфікації поряд з режимом роботи по індивідуальних лініях зв'язку.

В ході використання індивідуальних ліній зв'язку в повнозв'язаних топологіях кінцеві вузли повинні мати по одному порту на кожну лінію зв'язку. У зіркоподібних топологіях кінцеві вузли можуть підключатися індивідуальними лініями зв'язку до спеціального пристрою – комутатора. У глобальних мережах комутатори використовувалися вже на початковому етапі, а в локальних – з початку 1990-х років. Комутатори призводять до істотного подорожчання локальної мережі, тому поки що їхнє застосування обмежене, але в міру зниження вартості комутації цей підхід, можливо, витисне застосування поділюваних ліній зв'язку. Слід підкреслити, що індивідуальними в таких мережах є тільки лінії зв'язку між кінцевими вузлами й комутаторами мережі, а зв'язки між комутаторами залишаються поділюваними, оскільки по них передаються повідомлення різних кінцевих вузлів (рис. 1.7)

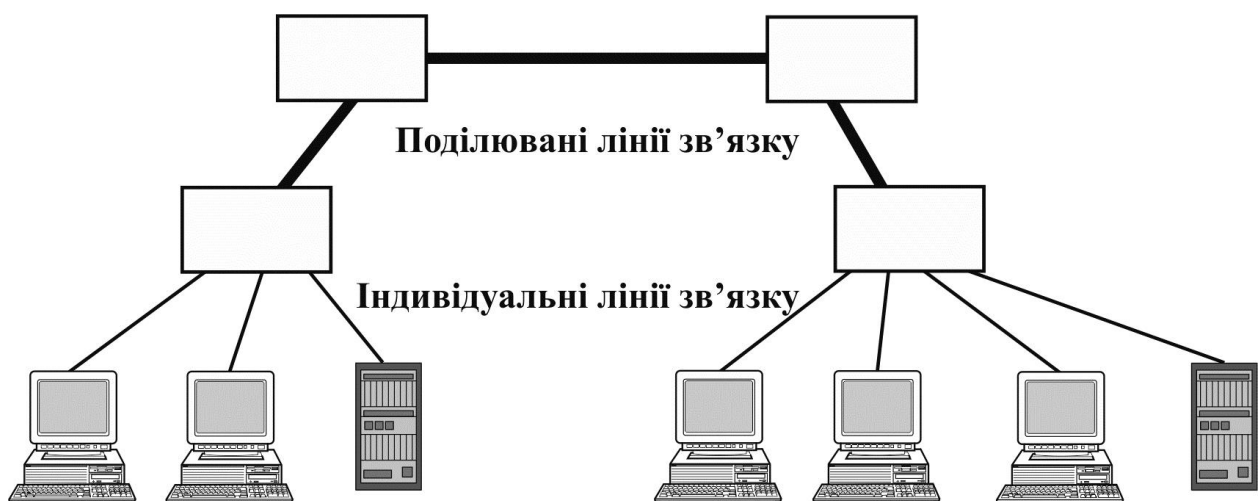


Рисунок 1.7 – Індивідуальні й поділювані лінії зв'язку в мережах на основі комутаторів

У глобальних мережах відмова від поділених ліній зв'язку пояснюється технічними причинами. Тут великі часові затримки поширення сигналів принципово обмежують застосованість техніки поділу лінії зв'язку. Комп'ютери можуть затратити більше часу на переговори про те, кому зараз можна використовувати лінію зв'язку, ніж безпосередньо на передачу даних по цій лінії зв'язку. Однак це не стосується ліній зв'язку типу «комутатор – комутатор». У даному випадку тільки два комутатори борються за доступ до лінії зв'язку, і це істотно спрощує задачу організації спільного використання лінії.

Адресація комп'ютерів

Додатковою проблемою, яку необхідно враховувати при об'єднанні трьох і більше комп'ютерів, є проблема їхньої адресації. До адреси вузла мережі й схемі його призначення можна пред'явити кілька вимог.

- Адреса має унікально ідентифікувати комп'ютер у мережі будь-якого масштабу.
- Схема призначення адрес має зводити до мінімуму ручну роботу адміністратора й імовірність дублювання адрес.
- Адреса повинна мати ієрархічну структуру, зручну для побудови великих мереж. Цю проблему добре ілюструють міжнародні поштові адреси, які дозволяють поштовій службі, що організує доставку листів між країнами, користуватися тільки назвою країни адресата й не враховувати назву його міста, а тим більше вулиці. У великих мережах, які складаються з багатьох тисяч вузлів, відсутність ієрархії адреси може призвести до великих витрат – кінцевим вузлам і комунікаційному обладнанню прийдеться оперувати з таблицями адрес, що складаються з тисяч записів.
- Адреса має бути зручною для користувачів мережі, а це означає, що вона повинна мати символічне подання, наприклад, Servers або www.cisco.com.
- Адреса повинна мати, за можливості, компактне подання, аби не перевантажувати пам'ять комунікаційної апаратури – мережних адаптерів, маршрутизаторів тощо.

Неважко помітити, що ці вимоги суперечливі – наприклад, адреса, яка має ієрархічну структуру, скоріше буде менш компактною, ніж неієрархічна (таку адресу часто називають «плоскою», тобто тією, що не має структури). Символьна ж адреса, швидше за все, потребуватиме більше пам'яті, ніж адреса-число.

Оскільки всі перераховані вимоги важко об'єднати в рамках якої-небудь однієї схеми адресації, то на практиці зазвичай використовується одразу кілька схем, так що комп'ютер одночасно має кілька адрес-імен. Кожна адреса використовується в тій ситуації, коли відповідний вид адресації найзручніший. А щоб не виникало плутанини, й комп'ютер завжди однозначно визначається своєю адресою, використовуються спеціальні допоміжні протоколи, які за адресою одного типу можуть визначити адреси інших типів.

Найбільше поширення отримали три схеми адресації вузлів:

- *Апаратні (hardware) адреси.* Ці адреси призначені для мережі невеликого або середнього розміру, тому вони не мають ієрархічної структури. Типовим представником адреси такого типу є адреса мережного адаптера локальної мережі. Така адреса зазвичай використовується тільки апаратурою, тому її намагаються зробити по можливості компактною й записують у вигляді двійкового або шістнадцяткового значення, наприклад, 0081005e24a8. Задання апаратних адрес зазвичай не потребує виконання ручної роботи, оскільки вони або вбудовуються в апаратуру компанією-виробником, або генеруються автоматично під час кожного нового запуску обладнання, причому унікальність адреси в межах мережі забезпечує обладнання. Крім відсутності ієрархії, використання апаратних адрес пов'язано ще з одним недоліком – під час заміни апаратури, наприклад, мережного адаптера, змінюється й адреса комп'ютера. Більше того, в ході установки декількох мережних адаптерів у комп'ютера з'являється кілька адрес, що не дуже зручно для користувачів мережі.

- *Символьні адреси або імена.* Ці адреси призначені для запам'ятовування людьми й тому зазвичай несуть значення навантаження. Символьні адреси легко використовувати як у невеликих, так і великих мережах. Для роботи у великих мережах символьне ім'я може мати складну ієрархічну структуру, наприклад ftp-arch1.ucl.ac.uk. Ця адреса говорить про те, що даний комп'ютер підтримує ftp-архів у мережі одного з коледжів Лондонського університету (University College London – ucl) і ця мережа належить до академічної галузі (ac) Internet Великобританії (United Kingdom – uk). Під час роботи в межах мережі Лондонського університету таке довге символьне ім'я явно надлишкове й замість нього зручно користуватися коротким символьним ім'ям, на роль якого добре підходить наймолодша складового повного імені, тобто ім'я ftp-arch1.

- *Числові складені адреси.* Символьні імена зручні для людей, але через змінний формат і потенційно великої довжини їхня передача по мережі не дуже економічна. Тому в багатьох випадках для роботи у великих мережах як адреси

вузлів використовуються числові складені адреси фіксованого й компактного форматів. Типовими представниками адрес цього типу є IP- та IPX-адреси. У них підтримується дворівнева ієрархія, адреса поділяється на старшу частину (номер мережі) та молодшу (номер вузла). Такий поділ дозволяє передавати повідомлення між мережами лише на підставі номера мережі, а номер вузла використовується тільки після доставки повідомлення в потрібну мережу; так само, як назва вулиці використовується листоношею тільки після того, як лист доставлений у потрібне місто. Останнім часом, щоб зробити маршрутизацію у великих мережах ефективнішою, пропонуються більш складні варіанти числової адресації, відповідно до яких адреса має три й більше складових. Такий підхід, зокрема, реалізований у версії протоколу IPv6, призначеного для роботи в мережі Internet. У сучасних мережах для адресації вузлів застосовуються, як правило, одночасно всі три наведені вище схеми. Користувачі адресують комп'ютери символьними іменами, які автоматично замінюються в повідомленнях, переданих мережею, на числові номери. За допомогою цих числових номерів повідомлення передаються з однієї мережі в іншу, а після доставки повідомлення в мережу призначення замість числового номера використовується апаратна адреса комп'ютера. Така схема характерна навіть для невеликих автономних мереж, де, здавалося б, вона явно надлишкова – це робиться для того, щоб при включенні цієї мережі у велику мережу не потрібно було міняти склад операційної системи.

Проблема встановлення відповідності між адресами різних типів, якою займається *служба дозволу імен*, може вирішуватися як повністю централізованими, так і розподіленими засобами. У випадку централізованого підходу в мережі виділяється один комп'ютер (сервер імен), у якому зберігається таблиця відповідності один одному імен різних типів, наприклад, символьних імен і числових номерів. Всі інші комп'ютери звертаються до сервера імен, щоб за символьним іменем знайти числовий номер комп'ютера, з яким необхідно обмінятися даними.

В ході використання другого, розподіленого підходу, кожен комп'ютер сам вирішує задачу встановлення відповідності між іменами. Наприклад, якщо користувач вказав для вузла призначення числовий номер, то перед початком передачі даних комп'ютер-відправник посилає всім комп'ютерам мережі повідомлення (таке повідомлення називається широкомовним) з проханням пізнати це числове ім'я. Всі комп'ютери, отримавши це повідомлення, порівнюють заданий номер зі своїм власним. Той комп'ютер, що виявив збіг,

посилає відповідь, яка містить його апаратну адресу, після чого стає можливим відправлення повідомлень локальною мережею.

Розподілений підхід гарний тим, що не припускає виділення спеціального комп'ютера, який до того ж часто вимагає ручного задання таблиці відповідності імен. Недоліком розподіленого підходу є необхідність широкомовних повідомлень, – такі повідомлення перевантажують мережу, оскільки вони потребують обов'язкової обробки всіма вузлами, а не тільки вузлом призначення. Тому розподілений підхід використовується лише в невеликих локальних мережах. У великих мережах розповсюдження широкомовних повідомлень всіма їхніми сегментами стає практично нереальним, оскільки їм характерний централізований підхід. Найвідомішою службою централізованого дозволу імен є служба Domain Name System (DNS) мережі Internet.

Ethernet – приклад стандартного вирішення мережних проблем

Розглянемо, як описані вище загальні підходи до вирішення найважливіших проблем побудови мереж втілені в найбільш популярній мережній технології – *Ethernet*.

Мережна технологія – це погоджений набір стандартних протоколів і програмно-апаратних засобів (наприклад, мережних адаптерів, драйверів, кабелів і рознімів), які їх реалізують, достатній для побудови обчислювальної мережі. Епітет «достатній» підкреслює та обставина, що цей набір є мінімальним набором засобів, за допомогою яких можна побудувати працездатну мережу. Можливо, цю мережу можна покращити, наприклад, за рахунок виділення в ній підмереж, що відразу зажадає крім протоколів стандарту Ethernet застосування протоколу IP, а також спеціальних комунікаційних пристроїв – маршрутизаторів. Покращена мережа буде, швидше за все, більш надійною та швидкодіючою, але за рахунок надбудов над засобами технології Ethernet, що склала базис мережі.

Термін «мережна технологія» найчастіше використовується в описаному вище вузькому сенсі, але іноді застосовується і його розширене тлумачення як будь-якого набору засобів і правил для побудови мережі, наприклад, «технологія наскрізної маршрутизації», «технологія створення захищеного каналу», «технологія IP-мереж».

Протоколи, на основі яких будується мережа певної технології (у вузькому сенсі), спеціально розроблялися для спільної роботи, тому від розробника мережі не потрібно додаткових зусиль з організації їхньої взаємодії.

Іноді мережні технології називають *базовими технологіями*, маючи на увазі, що на їхній основі будується базис будь-якої мережі. Прикладами базових мережних технологій можуть слугувати поряд з Ethernet такі відомі технології локальних мереж, як Token Ring і FDDI, або ж технології територіальних мереж X.25 і frame relay. Для отримання працездатної мережі в цьому випадку достатньо придбати програмні й апаратні засоби, які належать до однієї базової технології – мережні адаптери з драйверами, концентратори, комутатори, кабельну систему тощо, – і поєднати їх відповідно до вимог стандарту на дану технологію.

Стандарт Ethernet був прийнятий у 1980 році. Основний принцип, покладений в основу Ethernet, – *випадковий метод доступу* до поділюваного середовища передачі даних. Тут як середовище може використовуватися товстий або тонкий коаксіальний кабель, кручена пара, оптоволокно або радіохвилі (до речі, першою мережею, побудованою на принципі випадкового доступу до поділюваного середовища, була радіомережа Aloha Гавайського університету).

У стандарті Ethernet чітко зафіксована топологія електричних зв'язків. Комп'ютери підключаються до поділюваного середовища відповідно до типової структури «загальна шина» (рис. 1.8). За допомогою поділюваної в часі шини будь-які два комп'ютери можуть обмінюватися даними. Управління доступом до лінії зв'язку здійснюється спеціальними контролерами – мережними адаптерами Ethernet. Кожен комп'ютер, а точніше, кожен мережний адаптер, має унікальну адресу. Передача даних відбувається зі швидкістю 10 Мбіт/с. Ця величина є пропускнуою здатністю мережі Ethernet.

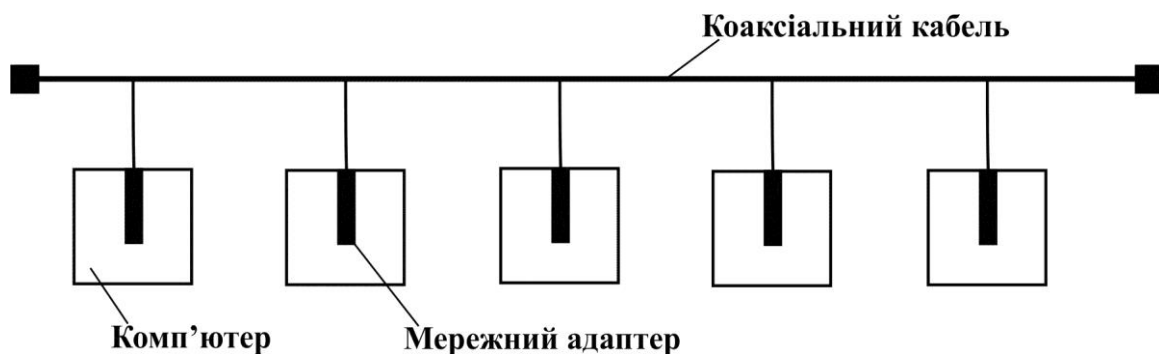


Рисунок 1.8 – Мережа Ethernet

Мережний адаптер

Суть випадкового методу доступу полягає в тому, що комп'ютер у мережі Ethernet може передавати дані мережею, тільки якщо мережа

вільна, тобто якщо ніякий інший комп'ютер у цей момент не займається обміном. Тому важливою частиною технології Ethernet є процедура визначення доступності середовища.

Після того, як комп'ютер переконався, що мережа вільна, він починає передавання, і при цьому «захоплює» середовище. Час монопольного використання поділюваного середовища одним вузлом обмежується часом передачі одного кадру. *Кадр* – це одиниця даних, якими обмінюються комп'ютери в мережі Ethernet. Кадр має фіксований формат і поряд з полем даних містить різну службову інформацію, наприклад, адреси отримувача й відправника.

Мережа Ethernet побудована так, що під час попадання кадру в поділюване середовище передачі даних всі мережні адаптери водночас починають приймати цей кадр. Усі вони аналізують адресу призначення, яка розташовується в одному з початкових полів кадру, і, якщо ця адреса збігається з їхньою власною адресою, кадр поміщається у внутрішній буфер мережного адаптера. Отже, комп'ютер-адресат отримує призначені йому дані.

Іноді може виникати ситуація, коли два або більше комп'ютерів водночас вирішують, що мережа вільна, і починають передавати інформацію. Така ситуація називається *колізією* і перешкоджає правильній передачі даних мережею. У стандарті Ethernet передбачено алгоритм виявлення й коректної обробки колізій. Імовірність виникнення колізії залежить від інтенсивності мережного трафіка.

Після виявлення колізії мережні адаптери, які намагалися передати свої кадри, припиняють передачу й після паузи випадкової тривалості намагаються знову отримати доступ до середовища й передати той кадр, що викликав колізію.

Головною перевагою мереж Ethernet є їхня економічність. Для побудови мережі досить мати по одному мережному адаптеру для кожного комп'ютера плюс один фізичний сегмент коаксіального кабелю потрібної довжини. Інші базові технології, наприклад, Token Ring, для створення навіть невеликої мережі вимагають наявності додаткового пристрою – концентратора.

Крім того, у мережах Ethernet реалізовані досить прості алгоритми доступу до середовища, адресації й передачі даних. Простота логіки роботи мережі веде до збільшення загальної довжини мережі. Повторювач передає сигнали, які приходять з одного сегмента мережі в інші її сегменти (рис. 1.9). Він дозволяє перебороти обмеження на довжину ліній зв'язку за рахунок покращення якості переданого сигналу – відновлення його потужності й амплітуди, покращення фронтів тощо.

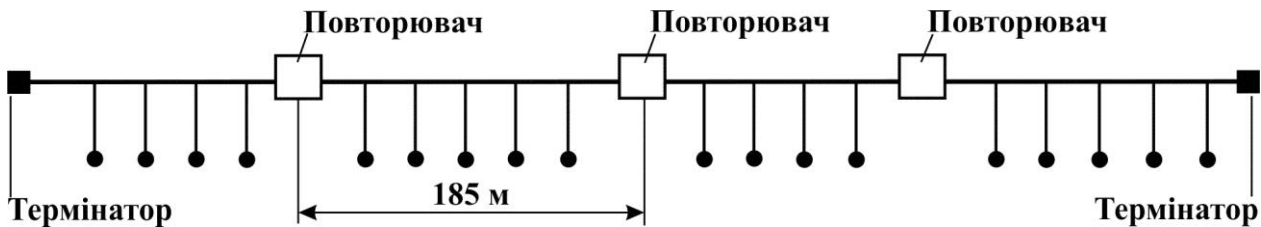


Рисунок 1.9 – Повторювач дозволяє збільшити довжину мережі Ethernet

Повторювач, що має кілька портів і з'єднує кілька фізичних сегментів, часто називають *концентратором* (*concentrator*) або *хабом* (*hub*). Ці назви (hub – основа, центр діяльності) відображають той факт, що в даному пристрої зосереджуються всі зв'язки між сегментами мережі.

Концентратори характерні практично для всіх базових технологій локальних мереж – Ethernet, ArcNet, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, IOOVG-AnyLAN.

Слід підкреслити, що в роботі концентраторів будь-яких технологій багато спільного – вони повторюють сигнали, які прийшли з одного зі своїх портів, на інших своїх портах. Відмінність в тому, на яких саме портах повторюються вхідні сигнали. Так, концентратор Ethernet повторює вхідні сигнали на всіх своїх портах, крім того, з якого сигнали надходять (рис. 1.10, а). А концентратор Token Ring (рис. 1.10, б) повторює вхідні сигнали, що надходять із деякого порту, тільки на одному порту – на тому, до якого підключений наступний у кільці комп'ютер.

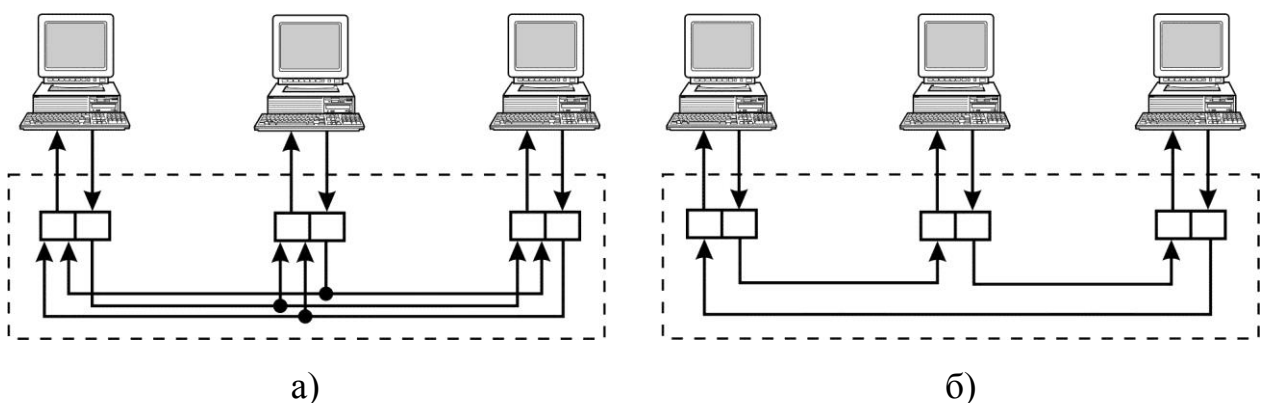


Рисунок 1.10 – Концентратори різних технологій

Концентратор завжди змінює фізичну топологію мережі, але при цьому залишає без зміни її логічну топологію.

Нагадаємо, що під фізичною топологією розуміють конфігурацію зв'язків, утворених окремими частинами кабелю, а під логічною –

конфігурацію інформаційних потоків між комп'ютерами мережі. У багатьох випадках фізична та логічна топології мережі збігаються. Наприклад, мережа, показана на рис. 1.11, а, має фізичну топологію кільце. Комп'ютери цієї мережі отримують доступ до кабелів кільця за рахунок передачі один одному спеціального кадра – маркера, причому цей маркер також передається послідовно від комп'ютера до комп'ютера в тому самому порядку, у якому комп'ютери утворюють фізичне кільце, тобто комп'ютер А передає маркер комп'ютеру В, комп'ютер В – комп'ютеру С й т.д. Мережа, показана на рис. 1.11, б, демонструє приклад розбіжності фізичної й логічної топології. Фізично комп'ютери з'єднані за топологією «загальна шина». Доступ до шини відбувається не за алгоритмом випадкового доступу, який застосовується в технології Ethernet, а шляхом передачі маркера в кільцевому порядку: від комп'ютера А – комп'ютеру В, від комп'ютера В – комп'ютеру С й т.д. Тут порядок передачі маркера вже не повторює фізичні зв'язки, а визначається логічним конфігуруванням драйверів мережних адаптерів. Ніщо не заважає налагодити мережні адаптери і їхні драйвери так, щоб комп'ютери утворили кільце в іншому порядку, наприклад: В, А, С. При цьому фізична структура мережі ніяк не змінюється.

Іншим прикладом розбіжності фізичної та логічної топологій мережі є вже розглянута мережа, на рис. 1.10, а. Концентратор Ethernet підтримує в мережі фізичну топологію «зірка». Однак логічна топологія мережі залишилася без змін – це спільна шина. Оскільки концентратор повторює дані, які прийшли з будь-якого порту, на всіх інших портах, то вони з'являються водночас на всіх фізичних сегментах мережі, як і в мережі з фізичною спільною шиною. Логіка доступу до мережі зовсім не міняється: усі компоненти алгоритму випадкового доступу – визначення незайнятості середовища, захоплення середовища, розпізнавання й відпрацьовування колізій – залишаються в силі.

Фізична структуризація мережі за допомогою концентраторів корисна не тільки для збільшення відстані між вузлами мережі, але й для підвищення її надійності. Наприклад, якщо будь який комп'ютер мережі Ethernet з фізичною спільною шиною через збій починає безупинно передавати дані по спільному кабелю, то вся мережа виходить з ладу, і для вирішення цієї проблеми залишається тільки один вихід – вручну від'єднати мережний адаптер цього комп'ютера від кабеля. У мережі Ethernet, побудованої з використанням концентратора, ця проблема може вирішуватися автоматично – концентратор відключає свій порт, якщо виявляє, що приєднаний до нього вузол

занадто довго монопольно займає мережу. Концентратор може блокувати некоректно працюючий вузол і в інших випадках, виконуючи роль деякого керуючого вузла.

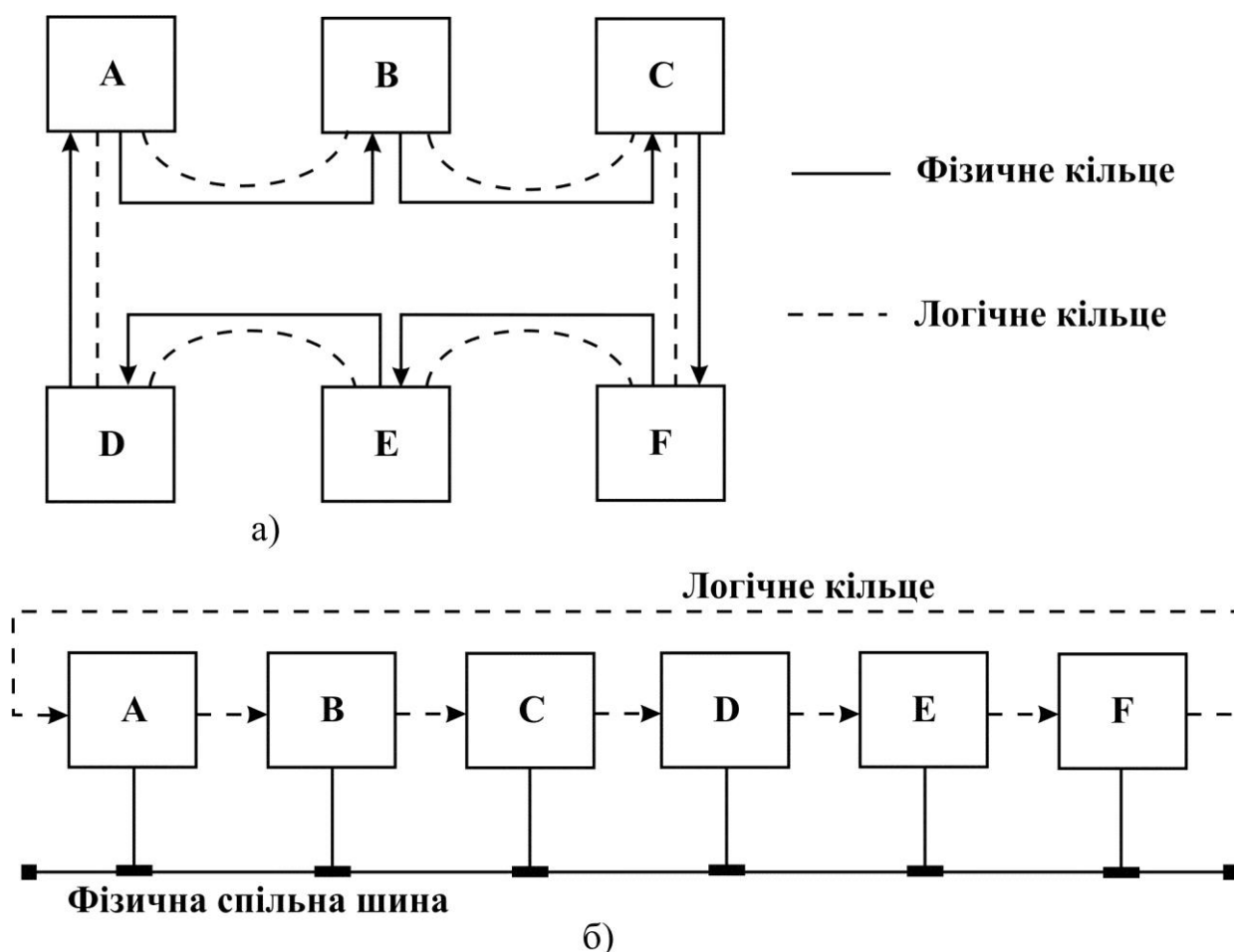


Рисунок 1.11 – Логічна й фізична топології мережі

Логічна структуризація мережі

Фізична структуризація мережі корисна в багатьох смислах, однак, у ряді випадків, що зазвичай стосуються мереж великого й середнього розміру, неможливо обійтися без логічної структуризації мережі. Найбільш важливою проблемою, яка не розв'язується шляхом фізичної структуризації, залишається проблема перерозподілу переданого трафіка між різними фізичними сегментами мережі.

У великій мережі природно виникає неоднорідність інформаційних потоків: мережа складається із множини підмереж робочих груп, відділів, філій підприємства й інших адміністративних утворень. Дуже часто найінтенсивніший обмін даними спостерігається між комп'ютерами, що належать до однієї підмережі, і тільки невелика частина обігів відбувається до ресурсів комп'ютерів, які перебувають поза локальними робочими групами.

(Донедавна таке співвідношення трафіків не піддавалося сумніву, і був навіть сформульований емпіричний закон «80/20», відповідно до якого у кожній підмережі 80% трафіка є внутрішнім і тільки 20% – зовнішнім.) На багатьох підприємствах є централізовані сховища корпоративних даних, які активно використовуються всіма співробітниками підприємства. Все це не могло не вплинути на розподіл інформаційних потоків. І тепер не рідкі ситуації, коли інтенсивність зовнішніх обігів вище інтенсивності обміну між «сусідніми» машинами. Але незалежно від того, у якій пропорції розподіляються зовнішній і внутрішній трафік, для підвищення ефективності роботи мережі неоднорідність інформаційних потоків необхідно враховувати.

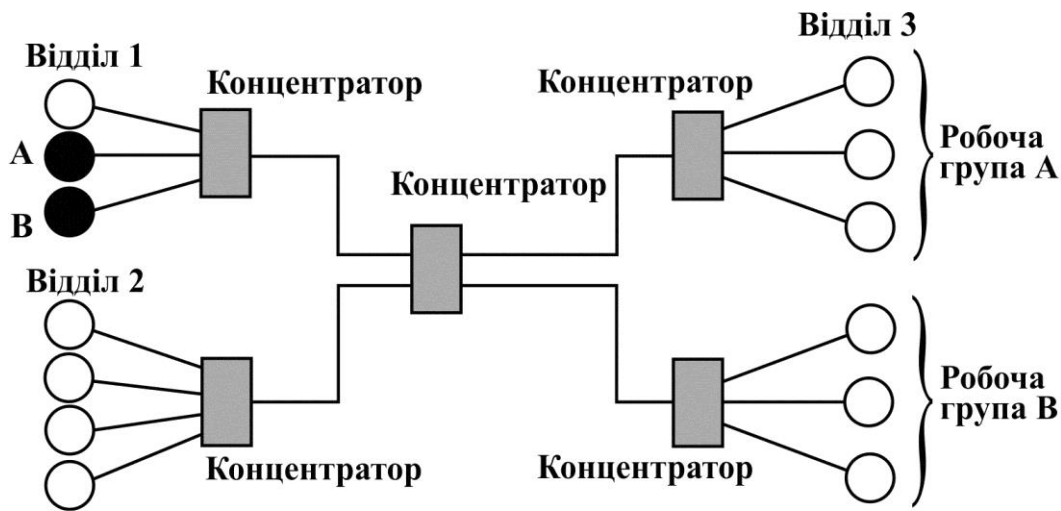
Мережа з типовою топологією (шина, кільце, зірка), в якій всі фізичні сегменти розглядаються як одне поділюване середовище, виявляється неадекватною структурі інформаційних потоків у великій мережі. Наприклад, у мережі зі спільною шиною взаємодія будь-якої пари комп'ютерів займає її на весь час обміну, тому при збільшенні кількості комп'ютерів у мережі шина стає вузьким місцем. Комп'ютери одного відділу змушені очікувати, коли закінчить обмін пара комп'ютерів іншого відділу, і це при тому, що необхідність у зв'язку між комп'ютерами двох різних відділів виникає набагато рідше й вимагає зовсім невеликої пропускну здатності.

Цей випадок ілюструє рис. 1.12, а. Тут показана мережа, побудована з використанням концентраторів. Нехай комп'ютер А, який перебуває в одній підмережі з комп'ютером В, посилає йому дані. Незважаючи на розгалужену фізичну структуру мережі, концентратори поширюють будь-який кадр по всіх її сегментах. Тому кадр, що посилається комп'ютером А комп'ютеру В, хоча й не потрібний комп'ютерам відділів 2 і 3, відповідно до логіки роботи концентраторів надходить на ці сегменти теж. І доти, доки комп'ютер В не отримає адресований йому кадр, жоден з комп'ютерів цієї мережі не зможе передавати дані.

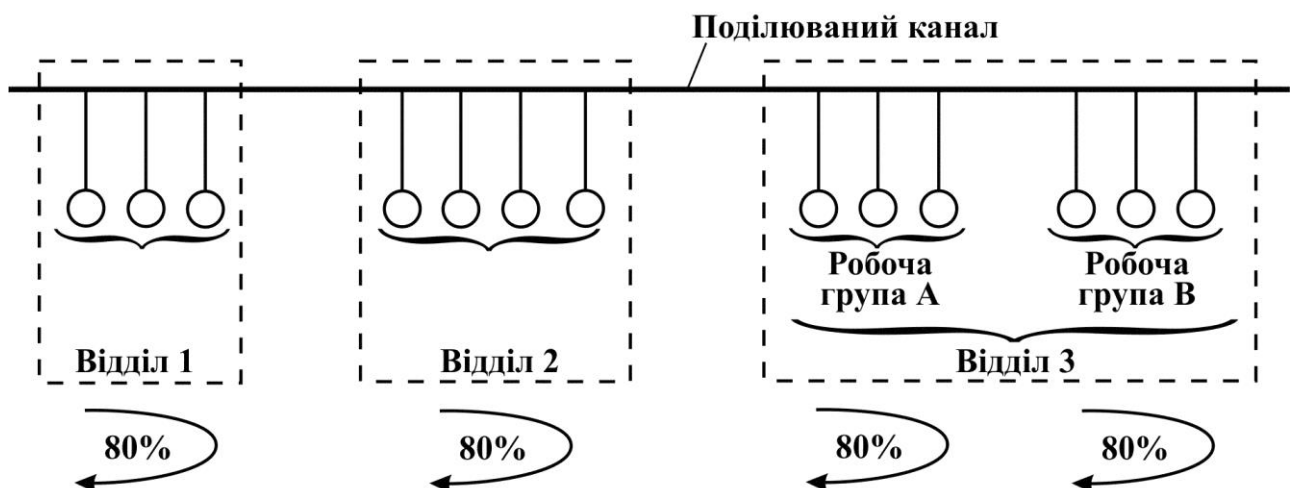
Така ситуація виникає через те, що логічна структура даної мережі залишилася однорідною, – вона ніяк не враховує збільшення інтенсивності трафіка всередині відділу й надає всім парам комп'ютерів рівні можливості з обміну інформацією (рис. 1.12, б).

Вирішення проблеми полягає у відмові від ідеї єдиного однорідного поділюваного середовища. Наприклад, у розглянутому вище прикладі бажано було б зробити так, щоб кадри, які передають комп'ютери відділу 1, виходили б за межі цієї частини мережі в тому і тільки в тому випадку, якщо ці кадри направлені якомусь комп'ютеру з інших відділів. З іншого боку, у мережу

кожного з відділів мають попадати ті, і тільки ті кадри, які адресовані вузлам цієї мережі. При такій організації роботи мережі її продуктивність істотно підвищиться, оскільки комп'ютери одного відділу не простоюватимуть у той час, коли обмінюються даними комп'ютери інших відділів.



а) Фізична структуризація за допомогою концентраторів



б) Логічна структура мережі залишилась без змін

Рисунок 1.12 – Протириччя між логічною структурою мережі
й структурою інформаційних потоків

Неважко помітити, що в запропонованому розв'язанні відмовилися від ідеї спільного поділюваного середовища в межах всієї мережі, хоча й залишили її в межах кожного відділу. Пропускна здатність ліній зв'язку між відділами не має збігатися із пропускнуою здатністю середовища усередині відділів. Якщо трафік між відділами становить тільки 20% трафіка усередині відділу (як ми вже зазначали, ця величина може бути іншою), то й пропускна здатність

ліній зв'язку та комунікаційного обладнання, що з'єднує відділи, може бути значно нижчою внутрішнього трафіка мережі відділу.

1.3 Логічна та фізична структуризація

Поширення трафіка, призначеного для комп'ютерів деякого сегмента мережі, тільки в межах цього сегмента, називається *локалізацією* трафіка. *Логічна структуризація* мережі – це процес розбивки мережі на сегменти з локалізованим трафіком.

Для логічної структуризації мережі використовуються такі комунікаційні пристрої, як мости, комутатори, маршрутизатори й шлюзи.

Mіст (bridge) ділить поділюване середовище передачі мережі на частини (які часто називаються логічними сегментами), передаючи інформацію з одного сегмента в інший тільки в тому випадку, якщо така передача дійсно необхідна, тобто якщо адреса комп'ютера призначення належить іншій підмережі. Тим самим міст ізолює трафік однієї підмережі від трафіка іншої, підвищуючи загальну продуктивність передачі даних у мережі. Локалізація трафіка не тільки заощаджує пропускну здатність, але й зменшує можливість несанкціонованого доступу до даних, оскільки кадри не виходять за межі свого сегмента і їх складніше перехопити зловмиснику.

На рис. 1.13 показана мережа, що була отримана з мережі із центральним концентратором (рис. 1.12) шляхом його заміни на міст.

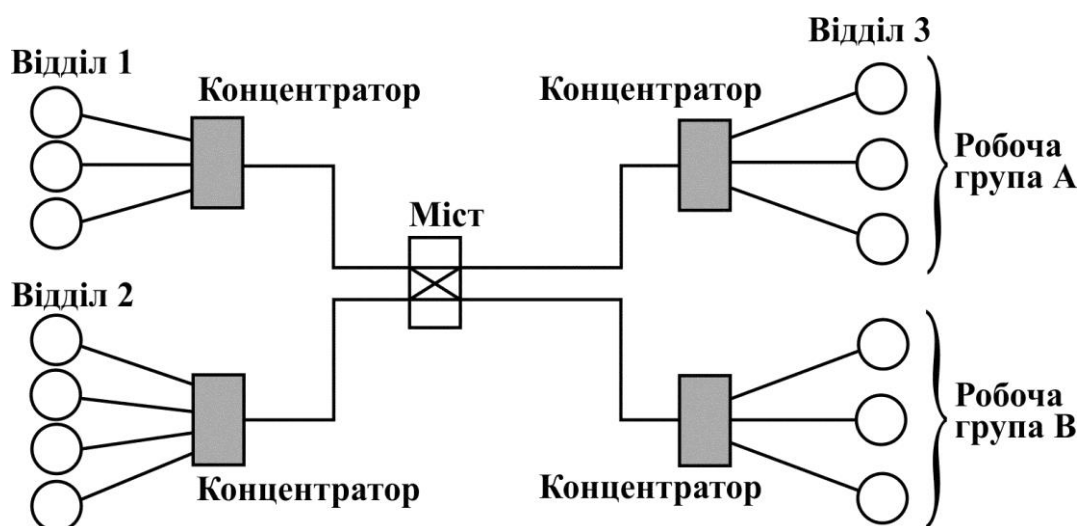


Рисунок 1.13 – Логічна структуризація мережі за допомогою мосту

Мережі 1-го й 2-го відділів складаються з окремих логічних сегментів, а мережа відділу 3 – із двох логічних сегментів. Кожний логічний сегмент

побудований на базі концентратора й має найпростішу фізичну структуру, утворену відрізками кабеля, які зв'язують комп'ютери з портами концентратора.

Мости використовують для локалізації трафіка апаратні адреси комп'ютерів. Це ускладнює розпізнавання приналежності того або іншого комп'ютера до певного логічного сегмента – сама адреса не містить ніякої інформації із цього приводу. Тому міст досить спрощено показує поділ мережі на сегменти – він запам'ятовує, через який порт на нього надійшов кадр даних від кожного комп'ютера мережі, і надалі передає кадри, призначені для цього комп'ютера, на цей порт. Точної топології зв'язків між логічними сегментами міст не знає. Через це застосування мостів призводить до значних обмежень на конфігурацію зв'язків мережі – сегменти мають бути з'єднані так, щоб у мережі не утворювалися замкнуті контури.

Комутатор (switch, switching hub) за принципом обробки кадрів нічим не відрізняється від мосту. Основна його відмінність в тому, що він є свого роду комунікаційним мультипроцесором, оскільки кожний його порт оснащений спеціалізованим процесором, що обробляє кадри за алгоритмом моста незалежно від процесорів інших портів. За рахунок цього загальна продуктивність комутатора зазвичай набагато вище продуктивності традиційного моста, який має один процесорний блок. Можна сказати, що комутатори – це мости нового покоління, які обробляють кадри у паралельному режимі.

Обмеження, пов'язані із застосуванням мостів і комутаторів – за топологією зв'язків та інші, – призвели до того, що в ряді комунікаційних пристроїв з'явився ще один тип обладнання – *маршрутизатор (router)*. Маршрутизатори надійніші та ефективніші, ніж мости, ізолюють трафік окремих частин мережі одну від одної. Маршрутизатори утворюють логічні сегменти за допомогою явної адресації, оскільки використовують не плоскі апаратні, а складові числові адреси. У цих адресах є поле номера мережі, так що всі комп'ютери, у яких значення цього поля однакове, належать до одного сегмента, який в цьому випадку називається *підмережею (subnet)*.

Крім локалізації трафіка маршрутизатори виконують ще багато інших корисних функцій. Так, маршрутизатори можуть працювати в мережі із замкнутими контурами, при цьому вони здійснюють вибір найбільш раціонального маршруту з декількох можливих. Мережа, подана на рис. 1.14, відрізняється від своєї попередниці (рис. 1.13) тим, що між підмережами відділів 1 і 2 прокладено додатковий зв'язок, який може використовуватися як для підвищення продуктивності мережі, так і для підвищення її надійності.

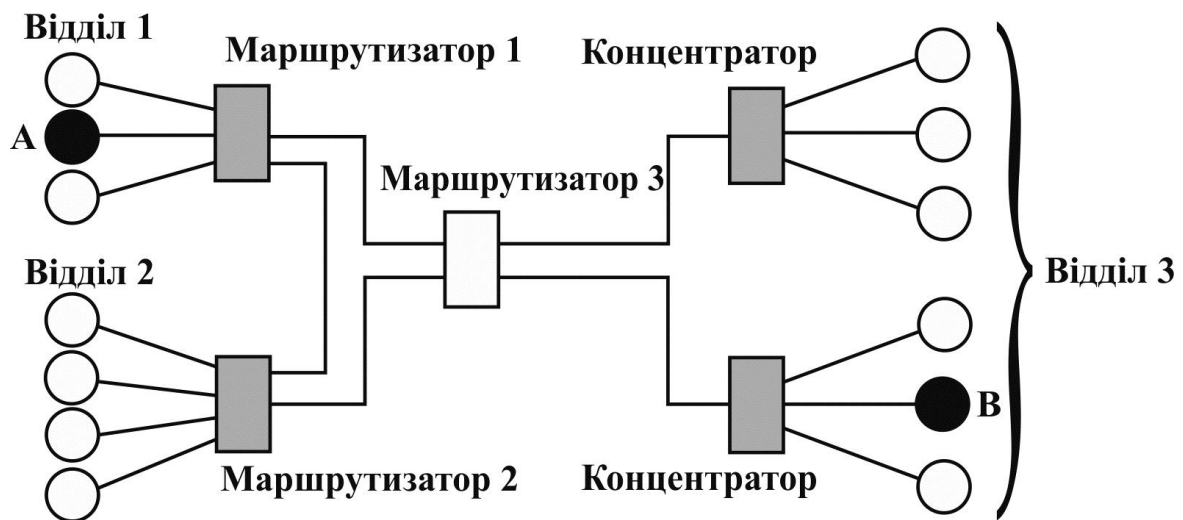


Рисунок 1.14 – Логічної структуризація мережі за допомогою маршрутизаторів

Іншою дуже важливою функцією маршрутизаторів є їхня здатність зв'язувати в єдину мережу підмережі, побудовані з використанням різних мережних технологій, наприклад, Ethernet і X.25.

Крім перерахованих пристроїв окремі частини мережі може з'єднувати *шлюз (gateway)*. Зазвичай основною причиною, за якої у мережі використовують шлюз, є необхідність об'єднати мережі з різними типами системного та прикладного програмного забезпечення, а не бажання локалізувати трафік. Проте, шлюз забезпечує й локалізацію трафіка як деякий побічний ефект.

Великі мережі практично ніколи не будуються без логічної структуризації. Для окремих сегментів і підмереж характерні типові однорідні топології базових технологій, і для їхнього об'єднання завжди використовується обладнання, яке забезпечує локалізацію трафіка, – мости, комутатори, маршрутизатори та шлюзи.

1.4 Контрольні запитання та завдання

1. Поясніть той факт, що глобальні мережі з'явилися раніше, ніж локальні.

2. Поясніть використання терміну «мережа» у таких висловах:

- мережа нашого підприємства включає *мережу* Ethernet і *мережу* Token Ring;
- маршрутизатор – це пристрій, який з'єднує *мережі*;
- щоб отримати вихід в Internet, необхідно отримати у постачальника послуг Internet номер *мережі*;

- останнім часом IP-мережі стають все більш розповсюдженими;
- гетерогенність корпоративної мережі призводить до того, що на перший план часто виходить проблема узгодження мереж.

3. Поясніть, чи всякий додаток, що виконується у мережі, можна назвати мережним?

4. Порівняйте, що спільного й у чому відмінність між взаємодією комп'ютерів у мережі й взаємодією комп'ютера з периферійним пристроєм.

5. Поясніть, як поділяються функції між мережним адаптером і його драйвером.

6. Поясніть значення термінів «клієнт», «сервер», «редиректор».

7. Назвіть головні недоліки повнозв'язної топології, а також топологій типу «спільна шина», «зірка», «кільце».

8. Назвіть топологію односегментної мережі Ethernet, побудованої на базі концентратора: «спільна шина» чи «зірка».

9. Оберіть, які з тверджень вірні:

- A. Поділ ліній зв'язку призводить до підвищення пропускної здатності каналу;
- B. Конфігурація фізичних зв'язків може збігатися з конфігурацією логічних зв'язків;
- C. Головною задачею служби дозволу імен є перевірка мережних імен і адрес на допустимість;
- D. Протоколи без встановлення з'єднань називаються також дейтаграмними протоколами.

Лекція 2

АРХІТЕКТУРА КОМП'ЮТЕРНИХ СИСТЕМ

2.1 Модель взаємодії відкритих систем

Сутність мережі – це з'єднання різного обладнання, та проблема сумісності є однією з найгостріших. Без прийняття всіма виробниками загальноприйнятих правил побудови обладнання прогрес у «будівництві» мереж був би неможливий. Тому весь розвиток комп'ютерної галузі остаточно відбито в стандартах – будь-яка нова технологія тільки тоді отримує «законний» статус, коли її сутність закріплюється у відповідному стандарті.

У комп'ютерних мережах ідеологічною основою стандартизації є багаторівневий підхід до розробки засобів мережної взаємодії. Саме на основі цього підходу була розроблена стандартна семирівнева модель взаємодії відкритих систем, що стала свого роду універсальною мовою мережних фахівців.

Багаторівневий підхід. Протокол. Інтерфейс. Стек протоколів

Організація взаємодії між пристроями в мережі є складною задачею. Як відомо, для вирішення складних задач використовується універсальний прийом – декомпозиція, тобто розбивка однієї складної задачі на декілька простіших задач-модулів (рис. 2.1).

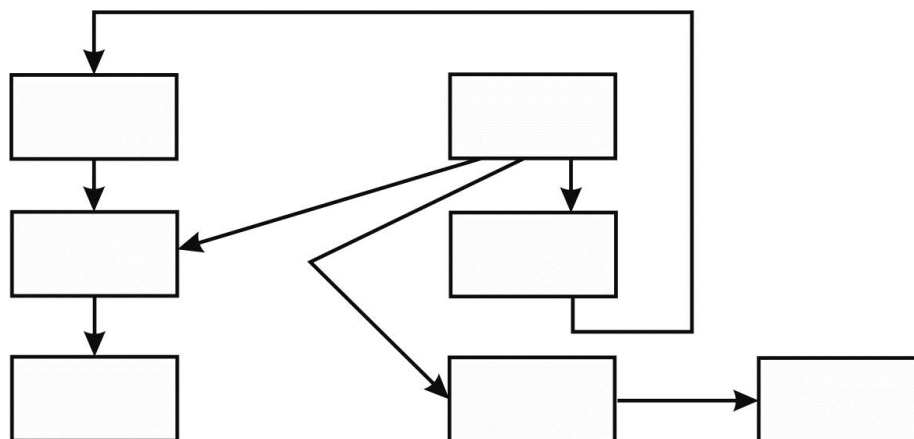


Рисунок 2.1 – Приклад декомпозиції задачі

Процедура декомпозиції містить чітке визначення функцій кожного модуля, який вирішує окрему задачу, та інтерфейсів між ними. У результаті досягається логічне спрощення задачі, а крім того, з'являється можливість модифікації окремих модулів без зміни іншої частини системи.

В ході декомпозиції часто використовують багаторівневий підхід. Він полягає у тому, що всю множину модулів розбивають на рівні. Рівні утворюють ієрархію, тобто є вищі та нижчі рівні (рис. 2.2). Множина модулів, які складають кожен рівень, сформована так, що для виконання своїх задач вони звертаються із запитом тільки до модулів, до яких безпосередньо примикає рівень, який лежить нижче. З іншого боку, результати роботи всіх модулів, що належать деякому рівню, можуть бути передані тільки модулям сусіднього рівня, який лежить вище. Така ієрархічна декомпозиція задачі припускає чітке визначення функції кожного рівня й інтерфейсів між рівнями. Інтерфейс визначає набір функцій, які нижчий рівень надає вищому. У результаті ієрархічної декомпозиції досягається відносна незалежність рівнів, а також і можливість їхньої легкої заміни.

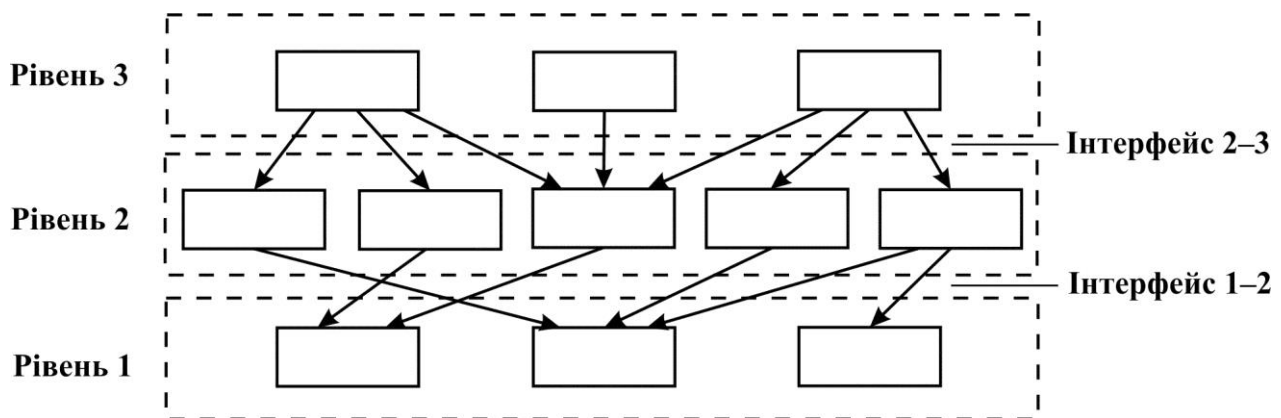


Рисунок 2.2 – Багаторівневий підхід – створення ієрархії задач

Засоби мережної взаємодії, зазвичай теж можуть бути подані у вигляді ієрархічно організованої множини модулів. При цьому модулі нижнього рівня можуть, наприклад, вирішувати всі питання, пов'язані з надійною передачею електричних сигналів між двома сусідніми вузлами. Модулі вищого рівня організують транспортування повідомлень у межах всієї мережі, користуючись для цього засобами згаданого нижчого рівня. А на верхньому рівні працюють модулі, які надають користувачам доступ до різних служб – файлової, друку тощо. Зазвичай, це тільки один із безлічі можливих варіантів поділу загальної задачі організації мережної взаємодії на часткові підзадачі.

Багаторівневий підхід до опису та реалізації функцій системи застосовується не тільки щодо мережних засобів. Така модель функціонування використовується, наприклад, у локальних файлових системах, коли запит, що надійшов, на доступ до файла послідовно оброблюється декількома програмними рівнями (рис. 2.3). Запит спочатку аналізується верхнім рівнем,

на якому здійснюється послідовний розбір складеного символьного імені файла й визначення унікального ідентифікатора файла. Наступний рівень знаходить за унікальним іменем усі основні характеристики файла: адресу, атрибути доступу тощо. Потім на нижчому рівні здійснюється перевірка прав доступу до цього файла, а далі, після розрахунку координат області файла, що містить необхідні дані, виконується фізичний обмін із зовнішнім пристроєм за допомогою драйвера диска.

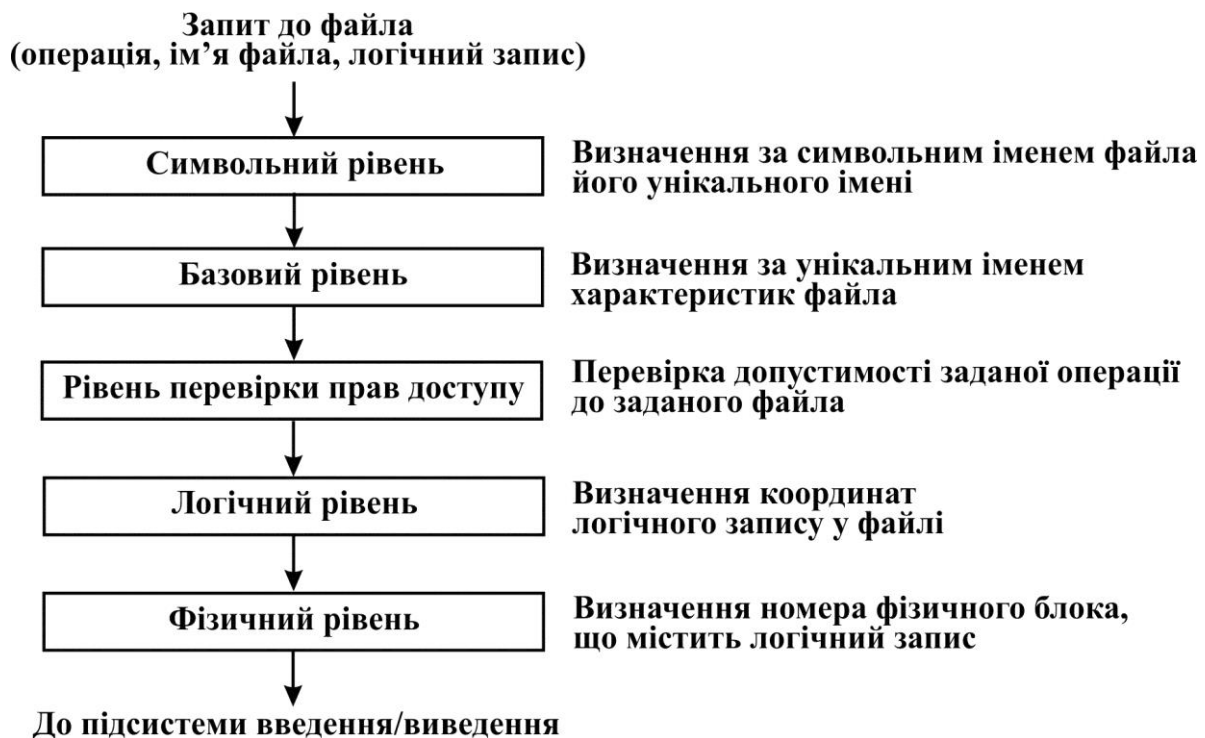


Рисунок 2.3 – Багаторівнева модель файлової системи

Багаторівневе подання засобів мережної взаємодії має свою специфіку, пов'язану з тим, що в процесі обміну повідомленнями беруть участь *дві* машини, тобто в цьому випадку необхідно організувати погоджену роботу двох «ієрархій». Під час передачі повідомлень обидва учасника мережного обміну мають прийняти безліч угод. Наприклад, вони мають погодити рівні та форму електричних сигналів, спосіб визначення довжини повідомлень, домовитися про методи контролю вірогідності тощо. Тобто угоди мають бути прийняті для всіх рівнів, починаючи від найнижчого (рівня передачі бітів) – до найвищого, який реалізує сервіс для користувачів мережі.

На рис. 2.4 показано модель взаємодії двох вузлів. Від кожної зі сторін засоби взаємодії подані чотирма рівнями. Процедура взаємодії цих двох вузлів може бути описана набором правил взаємодії кожної пари відповідних рівнів обох сторін, які беруть участь. Формалізовані правила, що визначають

послідовність і формат повідомлень, якими обмінюються мережні компоненти, що лежать на одному рівні, але в різних вузлах, називаються *протоколом*.

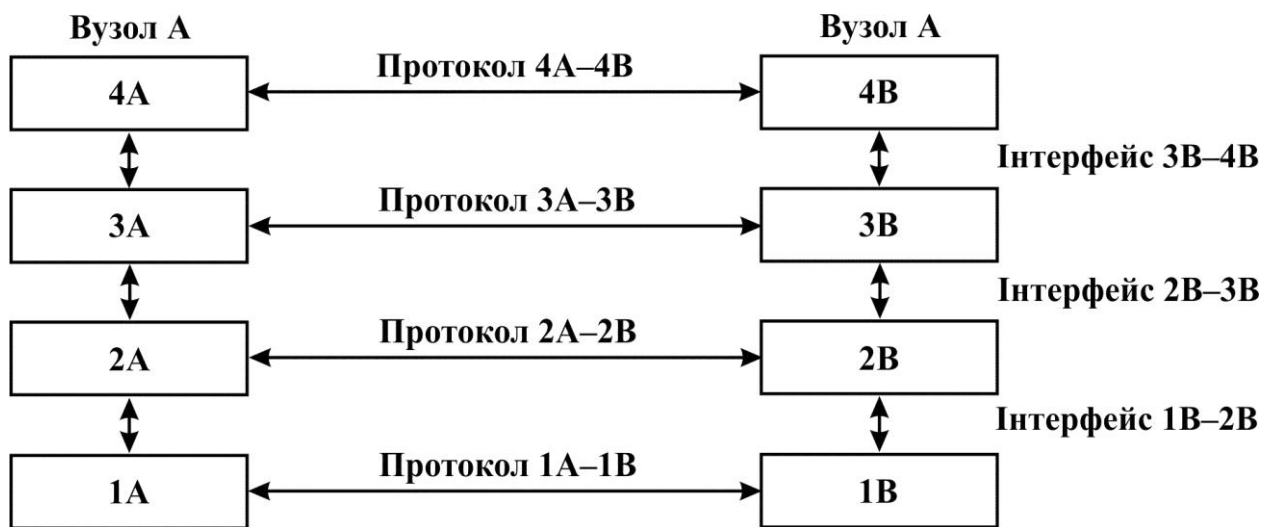


Рисунок 2.4 – Взаємодія двох вузлів

Модулі, які реалізують протоколи сусідніх рівнів і перебувають в одному вузлі, також взаємодіють один з одним відповідно до чітко визначених правил і за допомогою стандартизованих форматів повідомлень. Ці правила прийнято називати *інтерфейсом*. Інтерфейс визначає набір сервісів, що надається даним рівнем сусідньому рівню. По суті, протокол і інтерфейс виражають однакові поняття, але традиційно в мережах за ними закріпили різні галузі дії: протоколи визначають правила взаємодії модулів одного рівня в різних вузлах, а інтерфейси – модулів сусідніх рівнів в одному вузлі.

Засоби кожного рівня мають відпрацьовувати, по-перше, свій власний протокол, а по-друге, – інтерфейси з сусідніми рівнями.

Ієрархічно організований набір протоколів, достатній для організації взаємодії вузлів у мережі, називається *стеком комунікаційних протоколів*.

Комунікаційні протоколи можуть бути реалізовані як програмно, так і апаратно. Протоколи нижніх рівнів часто реалізуються комбінацією програмних і апаратних засобів, а протоколи верхніх рівнів – як правило, суто програмними засобами.

Програмний модуль, що реалізує деякий протокол, часто для стислості також називають «протоколом». При цьому співвідношення між протоколом – формально визначеною процедурою й протоколом – програмним модулем, що реалізує цю процедуру, аналогічно співвідношенню між алгоритмом розв’язання деякої задачі й програмою, яка вирішує цю задачу.

Зрозуміло, що один і той самий алгоритм може бути запрограмований з різним ступенем ефективності. Так само й протокол може мати кілька

програмних реалізацій. Саме тому під час порівняння протоколів слід враховувати не тільки логіку їхньої роботи, але і якість програмних рішень. Більш того, на ефективність взаємодії пристроїв у мережі впливає якість всієї сукупності протоколів, що складають стек, зокрема, наскільки раціонально розподілені функції між протоколами різних рівнів і наскільки добре визначені інтерфейси між ними.

Протоколи реалізуються не тільки комп'ютерами, але й іншими мережними пристроями – концентраторами, мостами, комутаторами, маршрутизаторами тощо. Дійсно, у загальному випадку зв'язок комп'ютерів у мережі здійснюється не прямо, а через різні комунікаційні пристрої. Залежно від типу пристрою, в ньому мають бути вбудовані засоби, які реалізують той чи інший набір протоколів.

Щоб ще раз пояснити поняття «протокол» і «інтерфейс», розглянемо приклад, що не стосується обчислювальних мереж, а саме обговоримо взаємодію двох підприємств А і В, пов'язаних між собою діловим співробітництвом. Між підприємствами існують численні домовленості та угоди, такі, як регулярні поставки продукції одного підприємства іншому. Відповідно до цієї домовленості начальник відділу продажів підприємства А регулярно на початку кожного місяця посилає офіційне повідомлення начальнику відділу закупівель підприємства В про те, який товар і в якій кількості може бути поставлено в цьому місяці. У відповідь на це повідомлення начальник відділу закупівель підприємства В посилає у відповідь заявку встановленого зразка на необхідну кількість продукції. Можливо, процедура взаємодії цих начальників включає додаткові узгодження, у будь-якому разі існує встановлений порядок взаємодії, який можна вважати «протоколом рівня начальників». Начальники посилають свої повідомлення й заявки через своїх секретарів. Порядок взаємодії начальника й секретаря відповідає поняттю міжрівневого інтерфейсу «начальник – секретар». На підприємстві А обмін документами між начальником і секретарем йде через спеціальну папку, а на підприємстві В начальник спілкується із секретарем по факсу. Отже інтерфейси «начальник – секретар» на цих двох підприємствах відрізняються.

Після того як повідомлення передані секретарям, начальників не хвилює, як ці повідомлення переміщатимуться далі – звичайною або електронною поштою, факсом або нарочним. Вибір способу передачі – це рівень компетенції секретарів, вони можуть вирішувати таке питання, не повідомляючи про це своїх начальників, тому що їхній протокол взаємодії пов'язаний тільки з передачею повідомлень, що надходять зверху, і не стосується змісту цих

повідомлень. На рис. 2.5 показано, що як протокол взаємодії «секретар-секретар» використовується обмін листами. При вирішенні інших питань начальники можуть взаємодіяти за іншими правилами-протоколами, але це не вплине на роботу секретарів, для яких не важливо, які повідомлення відправляти, а важливо, щоб вони дійшли до адресата. Отже, у цьому випадку ми маємо справу з двома рівнями – рівнем начальників і рівнем секретарів, і кожен з цих рівнів має власний протокол, який може бути змінений незалежно від протоколу іншого рівня. Ця незалежність протоколів один від одного і є привабливістю багаторівневого підходу.



Рисунок 2.5 – Приклад багаторівневої взаємодії підприємств

Модель OSI

З того, що протокол є угодою, прийнятою двома взаємодіючими об'єктами, у даному випадку двома працюючими в мережі комп'ютерами, зовсім не впливає, що він обов'язково є стандартним. Але на практиці під час реалізації мереж прагнуть використовувати стандартні протоколи. Це можуть бути фірмові, національні або міжнародні стандарти.

На початку 80-х років ряд міжнародних організацій по стандартизації – ISO, ITU-T й інші – розробили модель, що відіграла значну роль у розвитку мереж. Ця модель називається *моделлю взаємодії відкритих систем (Open System Interconnection, OSI)* або моделлю OSI. Модель OSI визначає різні рівні взаємодії систем, дає їм стандартні імена та вказує, які функції має виконувати кожен рівень. Модель OSI була розроблена на базі великого досвіду,

отриманого в ході створення комп'ютерних мереж, в основному глобальних, в 70-і роки. Повний опис цієї моделі займає більше 1000 сторінок тексту.

У моделі OSI (рис. 2.6) засоби взаємодії поділяються на сім рівнів: прикладний, представницький, сеансовий, транспортний, мережний, канальний та фізичний. Кожний рівень має справу з одним певним аспектом взаємодії мережних пристроїв.

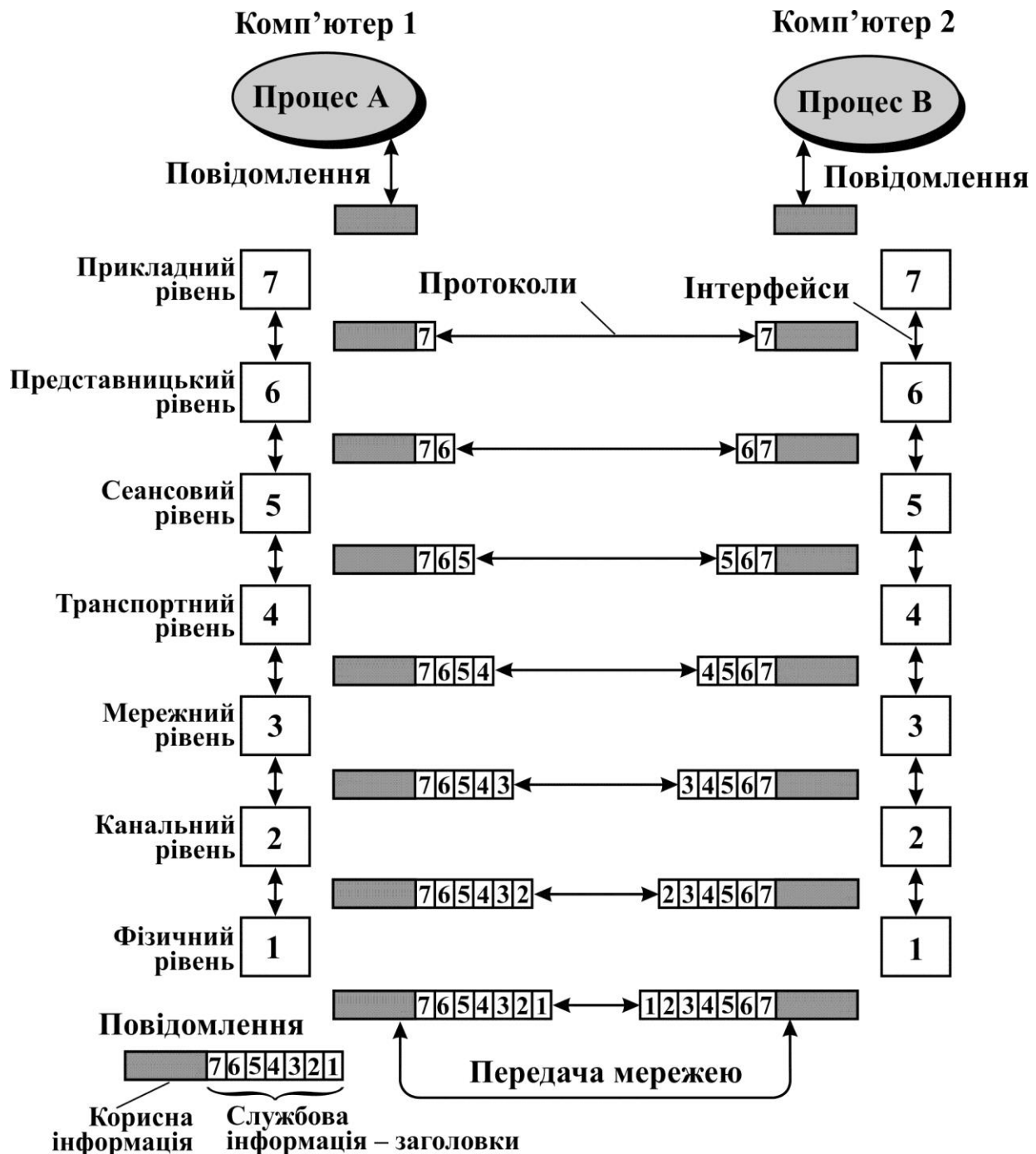


Рисунок 2.6 – Модель взаємодії відкритих систем ISO/OSI

Модель OSI описує тільки системні засоби взаємодії, реалізовані операційною системою, системними утилітами, системними апаратними

засобами. Модель не включає засоби взаємодії додатків кінцевих користувачів. Свої власні протоколи взаємодії додатки реалізують, звертаючись до системних засобів. Тому слід розрізняти рівень взаємодії додатків і прикладний рівень.

Також слід мати на увазі, що додаток може взяти на себе функції деяких верхніх рівнів моделі OSI. Наприклад, деякі СУБД мають вбудовані засоби віддаленого доступу до файлів. У цьому випадку додаток, виконуючи доступ до віддалених ресурсів, не використовує системну файлову службу; він обходить верхні рівні моделі OSI й звертається прямо до системних засобів, які відповідають за транспортування повідомлень по мережі і розташовуються на нижніх рівнях моделі OSI.

Отже, нехай додаток звертається із запитом до прикладного рівня, наприклад, до файлової служби. На підставі цього запиту програмне забезпечення прикладного рівня формує повідомлення стандартного формату. Звичайне повідомлення складається із заголовка й поля даних. Заголовок містить службову інформацію, яку необхідно передати через мережу прикладному рівню машини-адресата, щоб повідомити йому, яку роботу слід виконати. У нашому випадку заголовок має містити інформацію про місце знаходження файла й про тип операції, яку необхідно над ним виконати. Поле даних повідомлення може бути порожнім або містити будь-які дані, наприклад, ті, що необхідно записати у віддалений файл. Але для того, щоб доставити цю інформацію за призначенням, треба розв'язати ще багато задач, відповідальність за які несуть нижчі рівні.

Після формування повідомлення прикладний рівень направляє його вниз по стеку представницькому рівню. Протокол представницького рівня на підставі інформації, отриманої із заголовка прикладного рівня, виконує необхідні дії й додає до повідомлення власну службову інформацію – заголовок представницького рівня, у якому містяться вказівки для протоколу представницького рівня машини-адресата. Сформоване в результаті повідомлення передається вниз сеансовому рівню, що у свою чергу додає свій заголовок тощо. (Деякі реалізації протоколів містять службову інформацію не тільки на початку повідомлення у вигляді заголовка, але й наприкінці, у вигляді так званого «концевика».) Нарешті, повідомлення досягає нижнього, фізичного рівня, який, власне, й передає його по лініях зв'язку машині-адресату. До цього моменту повідомлення «обростає» заголовками всіх рівнів (рис. 2.7).

Коли повідомлення по мережі надходить на машину-адресат, воно приймається її фізичним рівнем і послідовно переміщується вгору з рівня

на рівень. Кожен рівень аналізує й опрацьовує заголовок свого рівня, виконуючи відповідні даному рівню функції, а потім вилучає цей заголовок і передає повідомлення вищому рівню.

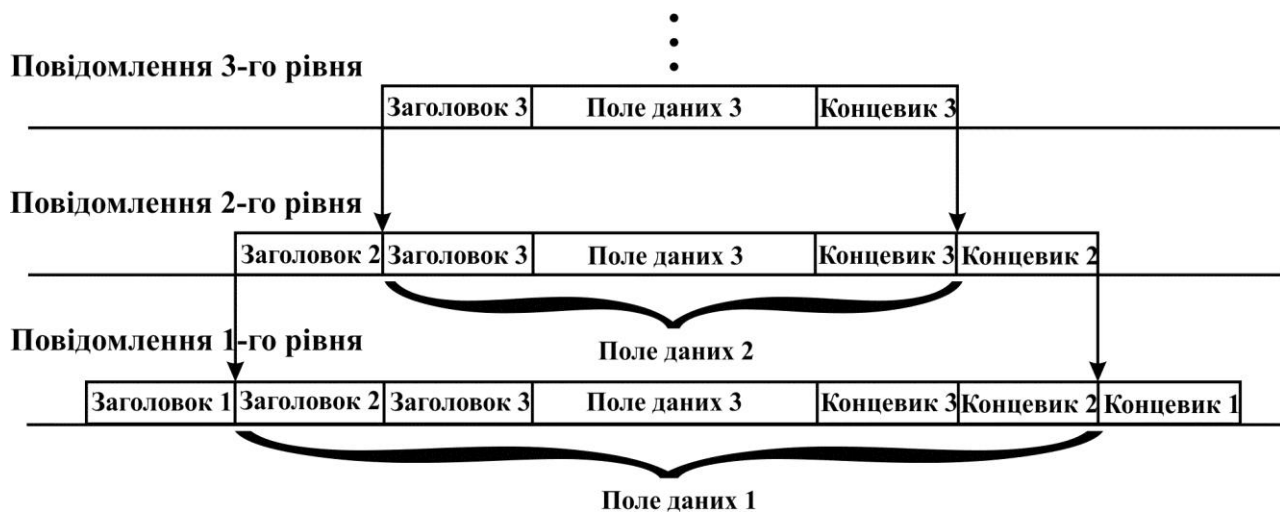


Рисунок 2.7 – Вкладеність повідомлень різних рівнів

Поряд з терміном повідомлення (*message*) існують й інші терміни, що застосовуються мережними фахівцями для позначення одиниць даних у процедурах обміну. У стандартах ISO для позначення одиниць даних, з якими мають справу протоколи різних рівнів, використовується загальна назва *протокольний блок даних (ProtocolData Unit, PDU)*. Для позначення блоків даних певних рівнів часто використовуються спеціальні назви: кадр (*frame*), пакет (*packet*), дейтаграма (*datagram*), сегмент (*segment*).

У моделі OSI розрізняють два основних типи протоколів. У протоколах із встановленням з'єднання (*connection-oriented*) перед обміном даними відправник і отримувач мають спочатку встановити з'єднання й, можливо, обрати деякі параметри протоколу, які вони використовуватимуть під час обміну даними. Після завершення діалогу вони мають розірвати це з'єднання. Телефон – це приклад взаємодії, заснований на встановленні з'єднання.

Друга група протоколів – протоколи без попереднього встановлення з'єднання (*connectionless*). Такі протоколи називаються також дейтаграмними протоколами. Відправник просто передає повідомлення, коли воно готово. Опускання листа в поштову скриньку – це приклад зв'язку без попереднього встановлення з'єднання. Під час взаємодії комп'ютерів використовуються протоколи обох типів.

2.2 Функції рівнів моделі OSI

Фізичний рівень

Фізичний рівень (Physical layer) працює з передачею бітів по фізичних каналах зв'язку, такими, як коаксіальний кабель, кручена пара, оптоволоконний кабель або цифровий територіальний канал. До цього рівня мають відношення характеристики фізичних середовищ передачі даних, такі як смуга пропускання, завадозахищеність, хвильовий опір тощо. На цьому самому рівні визначаються характеристики електричних сигналів, які передають дискретну інформацію, як от крутизна фронтів імпульсів, рівні напруги або струму переданого сигналу, тип кодування, швидкість передачі сигналів. Крім цього, тут стандартизуються типи рознімів і призначення кожного контакту.

Функції фізичного рівня реалізуються в усіх пристроях, підключених до мережі. З боку комп'ютера функції фізичного рівня виконуються мережним адаптером або послідовним портом.

Прикладом протоколу фізичного рівня може слугувати специфікація 10Base-T технології Ethernet, що визначає як використовуваний кабель неекрановану кручену пару категорії 3 з хвильовим опором 100 Ом, розніми RJ-45, максимальну довжину фізичного сегмента 100 метрів, манчестерський код для передачі даних у кабелі, а також деякі інші характеристики середовища та електричних сигналів.

Канальний рівень

На фізичному рівні просто пересилаються біти. При цьому не враховується, що в деяких мережах, у яких лінії зв'язку використовуються (розділяються) поперемінно декількома парами взаємодіючих комп'ютерів, фізичне середовище передачі може бути зайнятим. Тому однією з задач канального рівня (Data Link layer) є перевірка доступності середовища передачі. Іншою задачею канального рівня є реалізація механізмів виявлення та корекції помилок. Для цього на канальному рівні біти групуються в набори, що називаються *кадрами (frames)*. Канальний рівень забезпечує коректність передачі кожного кадра, поміщаючи спеціальну послідовність біт у початок і кінець кожного кадра, для його виділення, а також обчислює контрольну суму, обробляючи всі байти кадра певним способом і додаючи контрольну суму до кадра. Коли кадр приходить по мережі, одержувач знову обчислює контрольну суму отриманих даних і порівнює результат з контрольною сумою з кадра. Якщо вони збігаються, кадр вважається правильним і приймається. Якщо ж

контрольні суми не збігаються, то фіксується помилка. Канальний рівень може не тільки виявляти помилки, але й виправляти їх за рахунок повторної передачі ушкоджених кадрів. Необхідно зазначити, що функція виправлення помилок не обов'язкова для канального рівня, тому в деяких протоколах цього рівня вона відсутня, наприклад, в Ethernet і frame relay.

У протоколах канального рівня, які використовуються у локальних мережах, закладена певна структура зв'язків між комп'ютерами та способи їхньої адресації. Хоча канальний рівень і забезпечує доставку кадра між двома будь-якими вузлами локальної мережі, він це робить тільки в мережі з точно визначеною топологією зв'язків, саме тією топологією, для якої він був розроблений. До таких типових топологій, що підтримуються протоколами канального рівня локальних мереж, належать «спільна шина», «кільце» й «зірка», а також структури, отримані з них за допомогою мостів і комутаторів. Прикладами протоколів канального рівня є протоколи Ethernet, Token Ring, FDDI, IOVG-AnyLAN.

У локальних мережах протоколи канального рівня використовуються комп'ютерами, мостами, комутаторами й маршрутизаторами. У комп'ютерах функції канального рівня реалізуються спільними зусиллями мережних адаптерів та їхніх драйверів.

У глобальних мережах, які рідко мають регулярну топологію, канальний рівень часто забезпечує обмін повідомленнями тільки між двома сусідніми комп'ютерами, з'єднаними індивідуальною лінією зв'язку. Прикладами протоколів «точка-точка» (як часто називають такі протоколи) можуть слугувати широко розповсюджені протоколи PPP і LAP-B. У таких випадках для доставки повідомлень між кінцевими вузлами через усю мережу використовуються засоби мережного рівня. Саме так організовані мережі X.25. Іноді в глобальних мережах функції канального рівня в чистому вигляді виділити важко, оскільки в тому самому протоколі вони поєднуються з функціями мережного рівня. Прикладами такого підходу можуть слугувати протоколи технологій ATM і frame relay.

У цілому канальний рівень є досить потужним і закінченим набором функцій з пересилання повідомлень між вузлами мережі. У деяких випадках протоколи канального рівня виявляються самодостатніми транспортними засобами й можуть допускати роботу поверх них безпосередньо протоколів прикладного рівня або додатків, без залучення засобів мережного й транспортного рівнів. Наприклад, існує реалізація протокола управління мережею SNMP безпосередньо поверх Ethernet, хоча стандартно цей протокол

працює поверх мережного протоколу IP і транспортного протоколу UDP. Природно, що застосування такої реалізації буде обмеженим, – вона не підходить для складених мереж різних технологій, наприклад Ethernet і X.25, й навіть для такої мережі, у якій у всіх сегментах застосовується Ethernet, але між сегментами існують петлевидні зв'язки. А от у двосегментній мережі Ethernet, що об'єднана мостом, реалізація SNMP над канальним рівнем буде цілком працездатна.

Проте, для забезпечення якісного транспортування повідомлень у мережах будь-яких топологій і технологій функцій канального рівня виявляється недостатньо, тому в моделі OSI вирішення цієї задачі покладене на такі два рівні – мережний і транспортний.

Мережний рівень

Мережний рівень (Network layer) слугує для утворення єдиної транспортної системи, яка поєднує кілька мереж, причому ці мережі можуть використовувати зовсім різні принципи передачі повідомлень між кінцевими вузлами й мати довільну структуру зв'язків. Функції мережного рівня досить різноманітні. Почнемо їхній розгляд на прикладі об'єднання локальних мереж.

Протоколи канального рівня локальних мереж забезпечують доставку даних між будь-якими вузлами тільки в мережі з відповідною типовою топологією, наприклад, топологією ієрархічної зірки. Це досить жорстке обмеження, яке не дозволяє будувати мережі з розвинутою структурою, наприклад, мережі, що поєднують кілька мереж підприємства в єдину мережу, або високонадійні мережі, у яких існують надлишкові зв'язки між вузлами. Можна було б ускладнювати протоколи канального рівня для підтримки петлевидних надлишкових зв'язків, але принцип розподілу обов'язків між рівнями призводить до іншого вирішення. Щоб, з одного боку, зберегти простоту процедур передачі даних для типових топологій, а з іншого – допустити використання довільних топологій, вводиться додатковий мережний рівень.

На мережному рівні сам термін *мережа* наділяють специфічним значенням. У цьому випадку під мережею розуміють сукупність комп'ютерів, по'єднаних між собою відповідно до однієї зі стандартних типових топологій та таких, що використовують для передачі даних один із протоколів канального рівня, визначений для цієї топології.

Усередині мережі доставка даних забезпечується відповідним канальним рівнем, а от доставкою даних між мережами займається мережний рівень, що і підтримує можливість правильного вибору маршруту передачі повідомлення

навіть у тому випадку, коли структура зв'язків між складовими мережами має характер, відмінний від прийнятого в протоколах каналного рівня.

Мережі поєднуються між собою спеціальними пристроями, які називаються маршрутизаторами. *Маршрутизатор* – це пристрій, що збирає інформацію про топологію міжмережних поєднань і на її основі пересилає пакети мережного рівня в мережу призначення. Щоб передати повідомлення від відправника, який перебуває в одній мережі, отримувачу, який перебуває в іншій мережі, слід зробити деяку кількість *транзитних передач між мережами*, або *хопів* (від *hop* – стрибок), щоразу вибираючи маршрут, який підходить. Отже, маршрут є послідовністю маршрутизаторів, через які проходить пакет.

На рис. 2.8 показано чотири мережі, пов'язані трьома маршрутизаторами. Між вузлами А і В даної мережі пролягають два маршрути: перший через маршрутизатори 1 і 3, а другий – через маршрутизатори 1, 2 і 3.

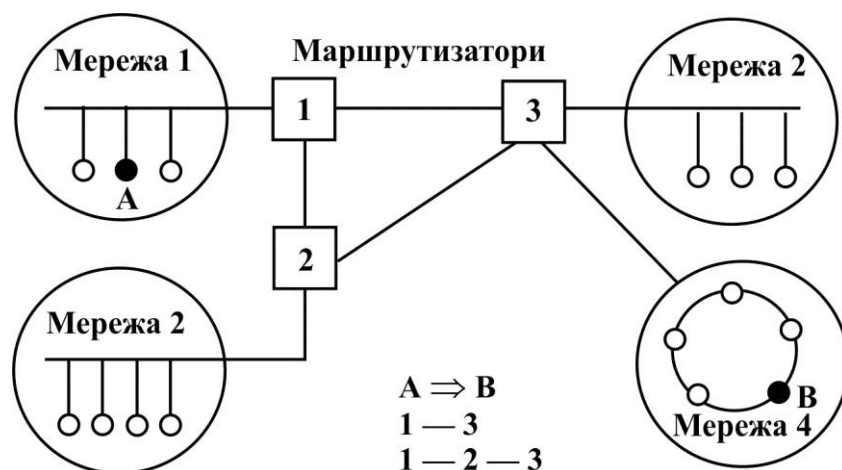


Рисунок 2.8 – Приклад складеної мережі

Проблема вибору найкращого шляху називається *маршрутизацією*, і її вирішення є однією з головних задач мережного рівня. Ця проблема ускладнюється тим, що самий короткий шлях не завжди найкращий. Часто критерієм при виборі маршруту є час передачі даних цим маршрутом; він залежить від пропускної здатності каналів зв'язку та інтенсивності трафіка, що може змінюватися із часом. Деякі алгоритми маршрутизації намагаються пристосуватися до зміни навантаження, у той час як інші приймають рішення на основі середніх показників за тривалий час. Вибір маршруту може здійснюватися й за іншими критеріями, наприклад, надійності передачі.

У загальному випадку функції мережного рівня ширше, ніж функції передачі повідомлень зв'язками з нестандартною структурою, які ми щойно розглянули на прикладі об'єднання декількох локальних мереж. Мережний

рівень вирішує також задачі узгодження різних технологій, спрощення адресації у великих мережах і створення надійних та гнучких бар'єрів на шляху небажаного трафіка між мережами.

Повідомлення мережного рівня прийнято називати *пакетами (packets)*. Під час організації доставки пакетів на мережному рівні використовується поняття «номер мережі». У цьому випадку адреса одержувача складається зі старшої частини – номера мережі й молодшої – номера вузла в цій мережі. Усі вузли однієї мережі повинні мати одну й ту саму старшу частину адреси, тому терміну «мережа» на мережному рівні можна дати й інше, більш формальне визначення: мережа – це сукупність вузлів, мережна адреса яких містить один той самий номер мережі.

На мережному рівні визначаються два типи протоколів. Перший тип – *мережні протоколи (routed protocols)* – реалізують просування пакетів по мережі. Зазвичай саме ці протоколи мають на увазі, коли говорять про протоколи мережного рівня. Однак часто до мережного рівня відносять й інший тип протоколів, які називаються протоколами обміну маршрутною інформацією або просто *протоколами маршрутизації (routing protocols)*. За допомогою цих протоколів маршрутизатори збирають інформацію про топологію міжмережних з'єднань. Протоколи мережного рівня реалізуються програмними модулями операційної системи, а також програмними й апаратними засобами маршрутизаторів.

На мережному рівні працюють протоколи ще одного типу, які відповідають за відображення адреси вузла, що використовується на мережному рівні, у локальну адресу мережі. Такі протоколи часто називають *протоколами дозволу адрес* – Address Resolution Protocol, ARP. Іноді вони належать не до мережного рівня, а до каналного, хоча тонкості класифікації не змінюють їхньої сутності.

Прикладами протоколів мережного рівня є протокол міжмережної взаємодії IP стека TCP/IP і протокол міжмережного обміну пакетами IPX стека Novell.

Транспортний рівень

На шляху від відправника до отримувача пакети можуть бути перекручені або загублені. Хоча деякі додатки мають власні засоби опрацювання помилок, існують і такі, що відразу віддають перевагу надійним з'єднанням. Транспортний рівень (Transport layer) забезпечує додаткам або верхнім рівням стека – прикладному й сеансовому – передачу даних з тим ступенем надійності, який їм потрібен. Модель OSI визначає п'ять класів сервісу, що надається транспортним рівнем. Ці види сервісу відрізняються якістю послуг: терміновістю, можливістю

відновлення перерваного зв'язку, наявністю засобів мультиплексування декількох з'єднань між різними прикладними протоколами через спільний транспортний протокол, а головне – здатністю до виявлення й виправлення помилок передачі, таких як перекручування, втрата й дублювання пакетів.

Вибір класу сервіса транспортного рівня визначається, з одного боку, тим, у якому ступені задача забезпечення надійності вирішується самими додатками й протоколами більш високих, ніж транспортний, рівнів, а з іншого, – цей вибір залежить від того, наскільки надійною є система транспортування даних у мережі, що забезпечується рівнями, розташованими нижче транспортного – мережним, канальним та фізичним. Так, наприклад, якщо якість каналів передачі зв'язку дуже висока й імовірність виникнення помилок, не виявлених протоколами більш низьких рівнів, невелика, то розумно скористатися одним з полегшених сервісів транспортного рівня, не обтяжених численними перевітками, квітуванням та іншими прийомами підвищення надійності. Якщо ж транспортні засоби нижніх рівнів від самого початку дуже ненадійні, то доцільно звернутися до найрозвиненішого сервісу транспортного рівня, який працює, використовуючи максимум засобів для виявлення й усунення помилок, – за допомогою попереднього встановлення логічного з'єднання, контролю доставки повідомлень по контрольних сумах і циклічній нумерації пакетів, встановлення тайм-аутів доставки тощо.

Як правило, всі протоколи, починаючи із транспортного рівня й вище, реалізуються програмними засобами кінцевих вузлів мережі – компонентами їхніх мережних операційних систем. Як приклад транспортних протоколів можна навести протоколи TCP і UDP стека TCP/IP і протокол SPX стека Novell.

Протоколи нижніх чотирьох рівнів узагальнено називають мережним транспортом або транспортною підсистемою, оскільки вони повністю вирішують задачу транспортування повідомлень із заданим рівнем якості в складених мережах з довільною топологією й різними технологіями. Інші три верхніх рівні вирішують задачі надання прикладних сервісів на основі наявної транспортної підсистеми.

Сеансовий рівень

Сеансовий рівень (Session layer) забезпечує управління діалогом: фіксує, яка зі сторін є активною в даний момент, надає засоби синхронізації. Останні дозволяють вставляти контрольні точки в довгі передачі, щоб у разі відмови можна було повернутися назад до останньої контрольної точки, а не починати все з початку. На практиці деякі додатки використовують сеансовий рівень, і він рідко

реалізується у вигляді окремих протоколів, хоча функції цього рівня часто поєднують із функціями прикладного рівня й реалізують в одному протоколі.

Представницький рівень

Представницький рівень (Presentation layer) має справу з формою подання переданої мережею інформації, не змінюючи при цьому її змісту. За рахунок рівня подання інформація, передана прикладним рівнем однієї системи, завжди зрозуміла прикладному рівню іншої системи. За допомогою засобів даного рівня протоколи прикладних рівнів можуть перебороти синтаксичні розходження в поданні даних або ж розходження в кодах символів, наприклад, кодів ASCII і EBCDIC. На цьому рівні може виконуватися шифрування й дешифрування даних, завдяки якому таємність обміну даними забезпечується відразу для всіх прикладних служб. Прикладом такого протоколу є протокол Secure Socket Layer (SSL), якій забезпечує секретний обмін повідомленнями для протоколів прикладного рівня стека TCP/IP.

Прикладний рівень

Прикладний рівень (Application layer) – це в дійсності просто набір різноманітних протоколів, за допомогою яких користувачі мережі отримують доступ до поділюваних ресурсів, таких як файли, принтери або гіпертекстові Web-сторінки, а також організують свою спільну роботу, наприклад, за допомогою протоколу електронної пошти. Одиниця даних, якою оперує прикладний рівень, зазвичай називається *повідомленням (message)*.

Існує дуже велика розмаїтість служб прикладного рівня. Наведемо як приклад декілька найпоширеніших реалізацій файлових служб: NCP в операційній системі Novell NetWare, SMB в Microsoft Windows NT, NFS, FTP і TFTP, що входять у стек TCP/IP.

Мережезалежні та мереженезалежні рівні

Функції всіх рівнів моделі OSI можуть бути віднесені до однієї з двох груп: або до функцій, які залежать від конкретної технічної реалізації мережі, або до функцій, орієнтованих на роботу з додатками.

Три нижніх рівні – фізичний, каналний та мережний – мережезалежні, тобто протоколи цих рівнів тісно пов'язані з технічною реалізацією мережі й використанням комунікаційним обладнанням. Наприклад, перехід на обладнання FDDI означає повну зміну протоколів фізичного та каналного рівнів у всіх вузлах мережі.

Три верхніх рівні – прикладний, представницький і сеансовий – орієнтовані на додатки й мало залежать від технічних особливостей побудови мережі. На протоколи цих рівнів не впливають які б то не було зміни в топології мережі, заміна обладнання або перехід на іншу мережну технологію. Так, перехід від Ethernet на високошвидкісну технологію IOGVG-AnyLAN не потребує ніяких змін у програмних засобах, які реалізують функції прикладного, представницького й сеансового рівнів.

Транспортний рівень є проміжним, він приховує всі деталі функціонування нижніх рівнів від верхніх. Це дозволяє розробляти додатки, які не залежать від технічних засобів безпосереднього транспортування повідомлень.

На рис. 2.9 показано рівні моделі OSI, на яких працюють різні елементи мережі.

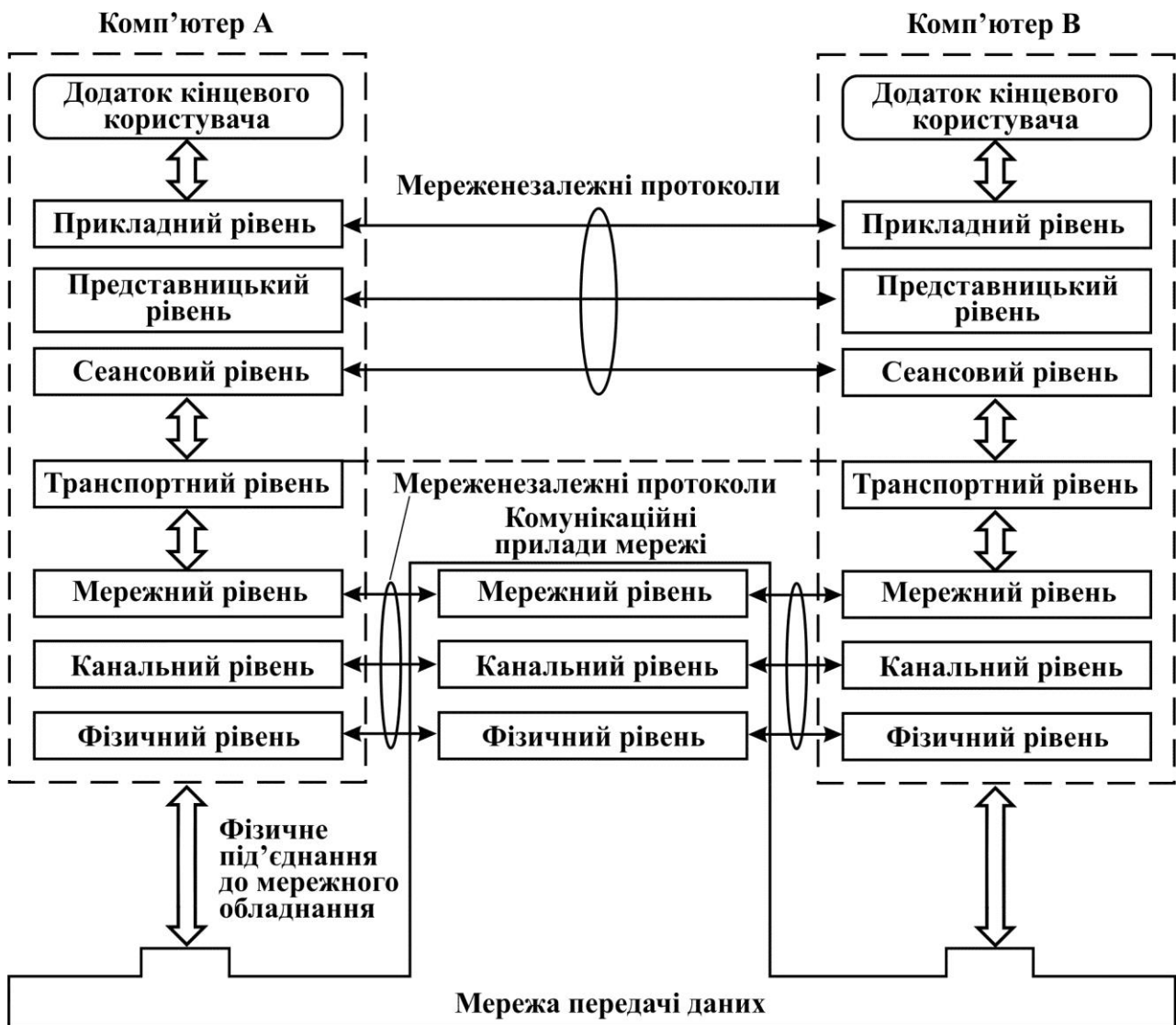


Рисунок 2.9 – Мереженезалежні та мереженезалежні рівні моделі OSI

Комп'ютер із встановленою на ньому мережною ОС взаємодіє з іншим комп'ютером за допомогою протоколів всіх семи рівнів. Цю взаємодію комп'ютери здійснюють опосередковано через різні комунікаційні пристрої: концентратори, модеми, мости, комутатори, маршрутизатори, мультиплексори. Залежно від типу комунікаційний пристрій може працювати або тільки на фізичному рівні (повторювач), або на фізичному й каналному (міст), або на фізичному, каналному та мережному, іноді захоплюючи й транспортний рівень (маршрутизатор). На рис. 2.10 показана відповідність функцій різних комунікаційних пристроїв рівням моделі OSI.

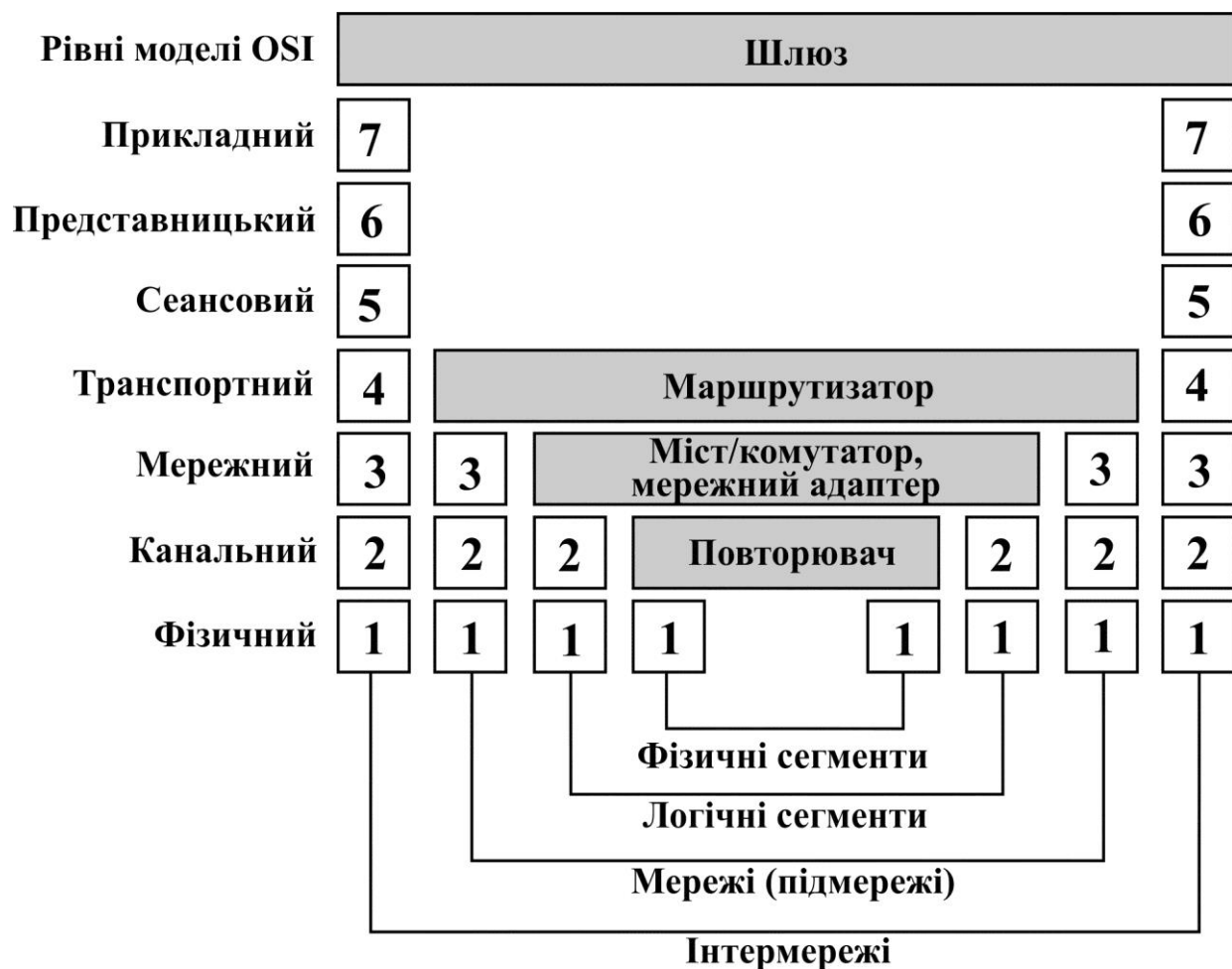


Рисунок 2.10 – Відповідність функцій різних пристроїв мережі рівням моделі OSI

Поняття «відкрита система»

Модель OSI, що витікає з її назви (Open System Interconnection), описує взаємозв'язки відкритих систем. Що є відкритою системою?

У широкому сенсі *відкритою системою* може бути названа будь-яка система (комп'ютер, обчислювальна мережа, ОС, програмний пакет,

інші апаратні й програмні продукти), яка побудована відповідно до відкритих специфікацій.

Модель OSI становить одну з багатьох моделей комунікацій. Ці моделі й пов'язані з ними стеки протоколів можуть відрізнятися кількістю рівнів, їхніми функціями, форматами повідомлень, службами, що підтримуються на верхніх рівнях, та іншими параметрами.

Нагадаємо, що під терміном «специфікація» (в обчислювальній техніці) розуміють формалізований опис апаратних або програмних компонентів, способів їхнього функціонування, взаємодії з іншими компонентами, умов експлуатації, обмежень і особливих характеристик. Зрозуміло, що не всяка специфікація є стандартом. У свою чергу, під відкритими специфікаціями розуміються опубліковані, загальнодоступні специфікації, що відповідають стандартам та прийняті в результаті досягнення згоди після всебічного обговорення всіма зацікавленими сторонами.

Використання під час розробки систем відкритих специфікацій дозволяє третім сторонам розробляти для цих систем різні апаратні або програмні засоби розширення та модифікації, а також створювати програмно-апаратні комплекси з продуктів різних виробників.

Для реальних систем повна відкритість є недосяжним ідеалом. Як правило, навіть у системах, які називаються відкритими, цьому визначенню відповідають лише деякі частини, що підтримують зовнішні інтерфейси. Наприклад, відкритість сімейства операційних систем Unix полягає, крім того, у наявності стандартизованого програмного інтерфейсу між ядром і додатками, що дозволяє легко переносити додатки із середовища однієї версії Unix у середовище іншої версії. Ще одним прикладом часткової відкритості є застосування в досить закритій операційній системі Novell NetWare відкритого інтерфейсу Open Driver Interface (ODI) для включення в систему драйверів мережних адаптерів незалежних виробників. Чим більше відкритих специфікацій використано під час розробки системи, тим більш відкритою вона є.

Модель OSI торкається тільки одного аспекту відкритості, а саме відкритості засобів взаємодії пристроїв, зв'язаних в обчислювальну мережу. Тут під відкритою системою розуміється мережний пристрій, готовий взаємодіяти з іншими мережними пристроями з використанням стандартних правил, які визначають формат, зміст і значення прийнятих повідомлень і повідомлень, що відправляються.

Якщо дві мережі побудовані з дотриманням принципів відкритості, то це дає такі переваги:

- можливість побудови мережі з апаратних і програмних засобів різних виробників, які дотримуються того самого стандарту;
- можливість безболісної заміни окремих компонентів мережі іншими, досконалішими, що дозволяє мережі розвиватися з мінімальними витратами;
- можливість легкого з'єднання однієї мережі з іншою;
- простота освоєння й обслуговування мережі.

Яскравим прикладом відкритої системи є міжнародна мережа Internet. Ця мережа розвивалася в повній відповідності з вимогами, пропонованими до відкритих систем. У розробці її стандартів брали участь тисячі фахівців – користувачів цієї мережі з різних університетів, наукових організацій і фірм-виробників обчислювальної апаратури й програмного забезпечення, що працюють у різних країнах. Сама назва стандартів, які визначають роботу мережі Internet – Request For Comments (RFC), що перекладається як «запит на коментарі», – показує публічний і відкритий характер прийнятих стандартів. У результаті мережа Internet зуміла об'єднати в собі найрізноманітніше обладнання й програмне забезпечення величезної кількості мереж, розкиданих по всьому світу.

Висновки

1. У комп'ютерних мережах ідеологічною основою стандартизації є багаторівневий підхід до розробки засобів мережної взаємодії.

2. Формалізовані правила, що визначають послідовність і формат повідомлень, якими обмінюються мережні компоненти, які лежать на одному рівні, але в різних вузлах, називаються протоколом.

3. Формалізовані правила, які визначають взаємодію мережних компонентів сусідніх рівнів одного вузла, називаються інтерфейсом. Інтерфейс визначає набір сервісів, що надається даним рівнем сусідньому рівню.

4. Ієрархічно організований набір протоколів, достатній для організації взаємодії вузлів у мережі, називається стеком комунікаційних протоколів.

5. Відкритою системою може бути названа будь-яка система, що побудована відповідно до загальнодоступних специфікацій, які відповідають стандартам і прийнятими в результаті публічного обговорення всіма зацікавленими сторонами.

6. Модель OSI стандартизує взаємодію відкритих систем. Вона визначає 7 рівнів взаємодії: прикладний, представницький, сеансовий, транспортний, мережний, каналний та фізичний.

7. Найважливішим напрямом стандартизації в галузі обчислювальних мереж є стандартизація комунікаційних протоколів. Найбільш популярними є стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA і OSI.

2.3 Контрольні запитання та завдання

1. Визначте функціональне призначення основних типів комунікаційного обладнання – повторювачів, концентраторів, мостів, комутаторів, маршрутизаторів.

2. Назвіть, у чому відмінність логічної структуризації мережі від фізичної?

3. Дайте пояснення: якщо всі комунікаційні пристрої в наведеному нижче фрагменті мережі (рис. 2.11) є концентраторами, то на яких портах з'явиться кадр, якщо його відправив комп'ютер А комп'ютеру В? Комп'ютеру С? Комп'ютеру D?

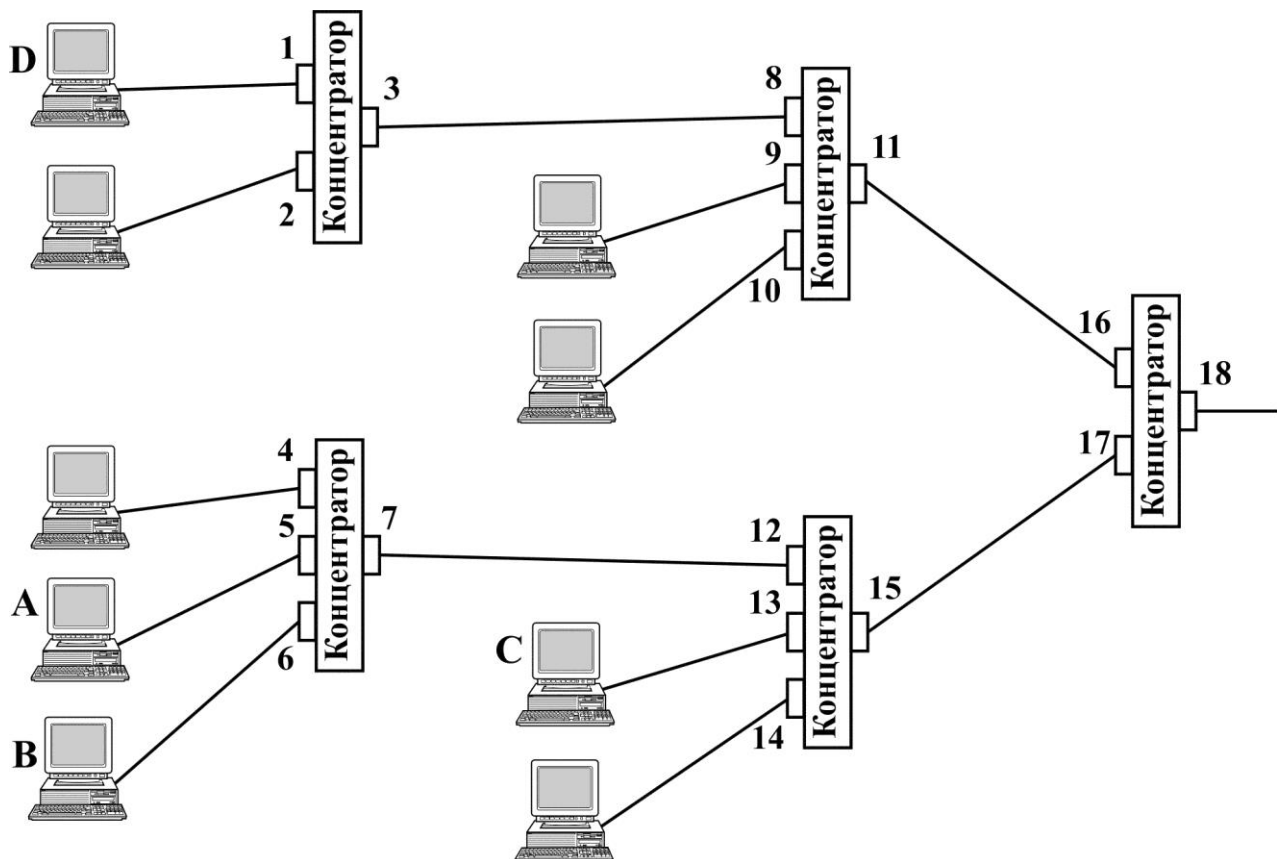


Рисунок 2.11 – Фрагмент мережі

4. Дайте пояснення: якщо в попередній вправі змінити умови та вважати, що всі комунікаційні пристрої є комутаторами, то на яких портах з'явиться кадр, відправлений комп'ютером А комп'ютеру В? Комп'ютеру С? Комп'ютеру D?
5. Поясніть, що таке «відкрита система»? Наведіть приклади закритих систем.
6. Поясніть відмінність у застосуванні термінів «протокол» і «інтерфейс» стосовно до багаторівневої моделі взаємодії пристроїв у мережі.
7. Визначте, що стандартизує модель OSI.
8. Визначте, що стандартизує стек OSI.
9. Поясніть, чому в моделі OSI сім рівнів.
10. Дайте стислий опис функцій кожного рівня й наведіть приклади стандартних протоколів для кожного рівня моделі OSI.
11. Визначте, чи є терміни «специфікація» і «стандарт» синонімами?
12. Назвіть організацію, яка розробила основні стандарти мереж Ethernet і Token Ring.

Лекція 3

СТАНДАРТНІ СТЕКИ КОМУНІКАЦІЙНИХ ПРОТОКОЛІВ

3.1 Стандартні стеки комунікаційних протоколів

Найважливішим напрямом стандартизації в галузі обчислювальних мереж є стандартизація комунікаційних протоколів. На сьогодні у мережах використовується велика кількість стеків комунікаційних протоколів. Найпопулярнішими стеками є: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA і OSI. Усі вони, крім SNA на нижніх рівнях – фізичному і каналному, – використовують ті самі добре стандартизовані протоколи Ethernet, Token Ring, FDDI і деякі інші, що дозволяють використовувати в усіх мережах ту саму апаратуру. Зате на верхніх рівнях усі стеки працюють за своїми власними протоколами. Часто ці протоколи не відповідають розбивці на рівні, які рекомендуються моделлю OSI. Зокрема, функції сеансового та представницького рівнів, як правило, об'єднані з прикладним рівнем. Така невідповідність пов'язана з тим, що модель OSI з'явилася як результат узагальнення вже існуючих і реально використовуваних стеків, а не навпаки.

Стек OSI

Слід чітко розрізняти модель OSI і стек OSI. У той час як модель OSI є концептуальною схемою взаємодії відкритих систем, стек OSI є набором цілком конкретних специфікацій протоколів. На відміну від інших стеків протоколів стек OSI повністю відповідає моделі OSI, він включає специфікації протоколів для всіх семи рівнів взаємодії, визначених у цій моделі. На нижніх рівнях стек OSI підтримує Ethernet, Token Ring, FDDI, протоколи глобальних мереж, X.25 і ISDN, – тобто використовує розроблені поза стеком протоколи нижніх рівнів, як і всі інші стеки. Протоколи мережного, транспортного та сеансового рівнів стека OSI специфіковані й реалізовані різними виробниками, але поширені поки що мало. Найпопулярнішими протоколами стека OSI є прикладні протоколи. До них належать: протокол передачі файлів ROTAM, протокол емуляції терміналу UTP, протоколи довідкової служби X.500, електронної пошти X.400 тощо.

Протоколи стека OSI відрізняє велика складність і неоднозначність специфікацій. Ці властивості стали результатом спільної політики розробників стека, які прагнули врахувати у своїх протоколах всі випадки життя та всі

існуючі технології й технології, що з'являються. До цього слід ще додати й наслідки великої кількості політичних компромісів, неминучих під час прийняття міжнародних стандартів з такого злободенного питання, як побудова відкритих обчислювальних мереж.

Через свою складність протоколи OSI вимагають великих витрат обчислювальної потужності центрального процесора, що робить їх найбільш зручними для потужних машин, а не для мереж персональних комп'ютерів.

Стек OSI – міжнародний, незалежний від виробників стандарт. Його підтримує уряд США у своїй програмі GOSIP, відповідно до якої всі комп'ютерні мережі, які встановлюються в урядових закладах США після 1990 року, мають або безпосередньо підтримувати стек OSI, або забезпечувати засобами для переходу на цей стек у майбутньому. Проте стек OSI більш популярний у Європі, ніж у США, оскільки в Європі залишилося менше старих мереж, які працюють за своїми власними протоколами. Більшість організацій поки що тільки планують перехід до стека OSI, і далеко не всі приступили до створення пілотних проектів. З тих, хто працює в цьому напрямі, можна назвати Військово-морське відомство США й мережу NFSNET. Одним з найбільших виробників, які підтримують OSI, є компанія AT&T, її мережа Stargroup повністю базується на цьому стеку.

Стек TCP/IP

Стек TCP/IP був розроблений з ініціативи Міністерства оборони США для зв'язку експериментальної мережі ARPAnet з іншими мережами як набір спільних протоколів для різноманітного обчислювального середовища. Великий внесок у розвиток стека TCP/IP, що отримав свою назву по популярних протоколах IP і TCP, вніс університет Берклі, реалізувавши протоколи стека у своїй версії ОС UNIX. Популярність цієї операційної системи призвела до широкого розповсюдження протоколів TCP, IP та інших протоколів стека. Сьогодні цей стек використовується для зв'язку комп'ютерів всесвітньої інформаційної мережі Internet, а також у величезній кількості корпоративних мереж.

Стек TCP/IP на нижньому рівні підтримує всі популярні стандарти фізичного й каналного рівнів: для локальних мереж – це Ethernet, Token Ring, FDDI, для глобальних – протоколи роботи на аналогових комутованих й виділених лініях SLIP, PPP, протоколи територіальних мереж X.25 і ISDN.

Основними протоколами стека, що дали йому назву, є протоколи IP і TCP. Ці протоколи в термінології моделі OSI належать до мережного й

транспортного рівнів відповідно. IP забезпечує просування пакета по складеній мережі, а TCP гарантує надійність його доставки.

За довгі роки використання в мережах різних країн й організацій стек TCP/IP увібрав велику кількість протоколів прикладного рівня. До них належать такі популярні протоколи, як протокол пересилання файлів FTP, протокол емуляції терміналу telnet, поштовий протокол SMTP, який використовується в електронній пошті мережі Internet, гіпертекстові сервіси служби WWW тощо.

Сьогодні стек TCP/IP є одним з найпоширеніших стеків транспортних протоколів обчислювальних мереж.

Стрімке зростання популярності Internet призвело й до змін у розстановці сил у світі комунікаційних протоколів – протоколи TCP/IP, на яких побудований Internet, стали швидко відтіснити безперечного лідера минулих років – стек IPX/SPX компанії Novell. Сьогодні у світі загальна кількість комп'ютерів, на яких установлений стек TCP/IP, зрівнялася із загальною кількістю комп'ютерів, на яких працює стек IPX/SPX, і це говорить про різкий перелом у відношенні адміністраторів локальних мереж до протоколів, які використовуються на настільних комп'ютерах, оскільки саме вони становлять переважну кількість світового комп'ютерного парку й саме на них раніше майже повсюди працювали протоколи компанії Novell, необхідні для доступу до файлових серверів NetWare. Процес становлення стека TCP/IP як «стека номер один» у будь-яких типах мереж триває, і зараз будь-яка промислова операційна система обов'язково включає програмну реалізацію цього стека у своєму комплекті поставки.

Хоча протоколи TCP/IP нерозривно пов'язані з Internet і кожний з багатомільйонної армadi комп'ютерів Internet працює на основі цього стека, існує велика кількість локальних, корпоративних та територіальних мереж, які безпосередньо не є частинами Internet, у яких також використовують протоколи TCP DP. Щоб відрізнити їх від Internet, ці мережі називають мережами TCP/IP або просто IP-мережами.

Оскільки стек TCP/IP від самого початку створювався для глобальної мережі Internet, він має багато особливостей, які дають йому перевагу перед іншими протоколами, коли мова заходить про побудову мереж, що включають глобальні зв'язки. Зокрема, дуже корисною властивістю, що робить можливим застосування цього протоколу у великих мережах, є його здатність фрагментувати пакети. Дійсно, велика складена мережа часто складається з мереж, побудованих на абсолютно різних принципах. У кожній із цих мереж

може бути встановлена власна величина максимальної довжини одиниці переданих даних (кадру). У такому випадку під час переходу з однієї мережі, що має більшу максимальну довжину, у мережу з меншою максимальною довжиною може виникнути необхідність поділу переданого кадру на декілька частин. Протокол IP-стека TCP/IP ефективно вирішує цю задачу.

Іншою особливістю технології TCP/IP є гнучка система адресації, що дозволяє більш просто, порівняно з іншими протоколами аналогічного призначення, включати в інтермережу мережі інших технологій.

Ця властивість також сприяє застосуванню стека TCP/IP для побудови великих гетерогенних мереж.

У стеку TCP/IP дуже економно використовуються можливості ширококомовних розсилань. Ця властивість абсолютно необхідна під час роботи на повільних каналах зв'язку, характерних для територіальних мереж.

Однак, як і завжди, за отримані переваги слід платити, і платою тут виявляються високі вимоги до ресурсів і складність адміністрування IP-мереж. Потужні функціональні можливості протоколів стека TCP/IP вимагають для своєї реалізації високих обчислювальних витрат. Гнучка система адресації й відмова від ширококомовних розсилань призводять до наявності в IP-мережі різних централізованих служб типу DNS, DHCP тощо. Кожна із цих служб спрямована на полегшення адміністрування мережі, у тому числі й на полегшення конфігурування обладнання, але в той же час сама вимагає пильної уваги з боку адміністраторів.

Можна приводити й інші доводи «за» й «проти» стека протоколів Internet, однак факт залишається фактом – сьогодні це самий популярний стек протоколів, який широко використовується як у глобальних, так і локальних мережах.

Стек IPX/SPX

Цей стек є оригінальним стеком протоколів фірми Novell, розробленим для мережної операційної системи NetWare ще на початку 80-х років. Протоколи мережного й сеансового рівнів Internetwork Packet Exchange (IPX) і Sequenced Packet Exchange (SPX), які дали назву стеку, є прямою адаптацією протоколів XNS фірми Xerox, розповсюджених у значно меншому ступені, ніж стек IPX/SPX. Популярність стека IPX/SPX безпосередньо пов'язана з операційною системою Novell NetWare, яка ще зберігає світове лідерство за кількістю встановлених систем, хоча останнім часом її популярність трохи знизилася й за темпами зростання вона відстає від Microsoft Windows NT.

Багато особливостей стека IPX/SPX обумовлено орієнтацією ранніх версій ОС NetWare (до версії 4.0) на роботу в локальних мережах невеликих розмірів, які складаються з персональних комп'ютерів зі скромними ресурсами. Зрозуміло, що для таких комп'ютерів компанії Novell потрібні були протоколи, на реалізацію яких була би необхідна мінімальна кількість оперативної пам'яті (обмеженої в IBM-сумісних комп'ютерах під управлінням MS-DOS обсягом 640 Кбайт) і які б швидко працювали на процесорах невеликої обчислювальної потужності. У результаті протоколи стека IPX/SPX донедавна добре працювали в локальних мережах і не дуже – у великих корпоративних мережах, оскільки вони занадто перевантажували повільні глобальні зв'язки широкомовними пакетами, які інтенсивно використовуються декількома протоколами цього стека (наприклад, для встановлення зв'язку між клієнтами й серверами). Ця обставина, а також той факт, що стек IPX/SPX є власністю фірми Novell і на його реалізацію потрібно отримувати ліцензію (тобто відкриті специфікації не підтримувалися), довгий час обмежували поширеність його тільки мережами NetWare. Однак з моменту випуску версії NetWare 4.0 Novell внесла й продовжує вносити у свої протоколи серйозні зміни, спрямовані на їхню адаптацію для роботи в корпоративних мережах. Зараз стек IPX/SPX реалізований не тільки в NetWare, але й у декількох інших популярних мережних ОС, наприклад SCO UNIX, Sun Solaris, Microsoft Windows NT.

На рис. 3.1 показана відповідність деяких, найбільш популярних протоколів рівням моделі OSI. Часто ця відповідність досить умовна, оскільки модель OSI – це тільки керівництво до дії, причому досить загальне, а конкретні протоколи розроблялися для вирішення специфічних задач, і багато які з них з'явилися до розробки моделі OSI. У більшості випадків розробники стеків віддавали перевагу швидкості роботи мережі на шкоду модульності – жоден стек, крім стека OSI, не розбито на сім рівнів. Найчастіше в стеку явно виділяються 3–4 рівні: рівень мережних адаптерів, у якому реалізуються протоколи фізичного й канального рівнів, мережний рівень, транспортний рівень і рівень служб, що вбирає в себе функції сеансового, представницького і прикладного рівнів.

Модель OSI	IBM/Microsoft	TCP/IP	Novell	Стек OSI
Прикладний	SMB	Telnet, FTP, SNMP, SMTP, WWW	NCP, SAP	X.400, X.500, FTAM
Представницький				Представницький протокол OSI
Сеансовий	NetBIOS	TPC	SPX	Сеансовий протокол OSI
Транспортний				Транспортний протокол OSI
Мережний		IP, RIP, OCF	IPX, RIP, NLSP	E8-E8, IS-IS
Канальний	802.3 (Ethernet), 8.2.5 (Token Ring), FDDI, Fast Ethernet, SLIP, 100VG-AnyLAN, X.25, ATM, LAP-B, LAP-D, PPP			
Фізичний	Коаксіал, екранована та неекранована кручена пара, оптоволокно, радіохвилі			

Рисунок 3.1 – Відповідність популярних стеків протоколів моделі OSI

3.2 Вимоги, які висуваються до обчислювальних мереж

Головною вимогою до мереж є виконання мережею її основної функції – забезпечення користувачам потенційної можливості доступу до поділюваних ресурсів всіх комп'ютерів, об'єднаних у мережу. Всі інші вимоги – надійність, продуктивність, сумісність, керованість, захищеність, розширюваність і масштабованість – пов'язані з якістю виконання цієї основної задачі.

Хоча всі ці вимоги досить важливі, часто поняття «якість обслуговування» (Quality of Service, QoS) комп'ютерної мережі трактується вузьче – воно містить лише дві найважливіші характеристики мережі – продуктивність і надійність.

Незалежно від обраного показника якості обслуговування мережі існують два підходи до його забезпечення.

Перший підхід, мабуть, здасться найприроднішим з точки зору користувача мережі. Він полягає в тому, що мережа (точніше, персонал, що її обслуговує) гарантує користувачу дотримання деякої числової величини показника якості обслуговування. Наприклад, мережа може гарантувати користувачу А, що кожний з його пакетів, посланих користувачу В, буде

затриманий мережею не більш, ніж на 150 мс. Або, що середня пропускна здатність каналу між користувачами А і В не буде нижчою 5 Мбіт/с, при цьому канал буде дозволяти пульсації трафіка в 10 Мбіт на інтервалах часу не більше двох секунд. Технології frame relay і АТМ дозволяють будувати мережі, які гарантують якість обслуговування по продуктивності.

Другий підхід полягає в тому, що мережа обслуговує користувачів відповідно до їх пріоритетів. Тобто якість обслуговування залежить від ступеня привілейованості користувача або групи користувачів, до якої він належить. Якість обслуговування в цьому випадку не гарантується, а гарантується лише рівень привілеїв користувача. Таке обслуговування називається обслуговуванням best effort – з найбільшим старанням. Мережа намагається за можливості більш якісно обслужити користувача, але нічого при цьому не гарантує. За таким принципом працюють, наприклад, локальні мережі, побудовані на комутаторах з пріоритезацією кадрів.

Продуктивність

Потенційно висока продуктивність – це одне з основних властивостей розподілених систем, до яких належать комп'ютерні мережі. Ця властивість забезпечується можливістю розпаралелювання робіт між декількома комп'ютерами мережі. На жаль, цю можливість не завжди вдається реалізувати.

Існує кілька основних характеристик продуктивності мережі:

- час реакції;
- пропускна здатність;
- затримка передачі й варіація затримки передачі.

Час реакції мережі є інтегральною характеристикою продуктивності мережі з точки зору користувача. Саме цю характеристику має на увазі користувач, коли говорить: «Сьогодні мережа працює повільно».

У загальному випадку час реакції визначається як інтервал часу між виникненням запиту користувача до будь-якої мережної служби й отриманням відповіді на цей запит.

Вочевидь, що значення цього показника залежить від типу служби, до якої звертається користувач, від того, який користувач і до якого сервера звертається, а також від поточного стану елементів мережі – завантаженості сегментів, комутаторів і маршрутизаторів, через які проходить запит завантаженості сервера тощо. Тому має сенс використовувати також і середньозважену оцінку часу реакції мережі, усереднюючи цей показник по користувачах, серверах і часу доби (від якого значною мірою залежить завантаження мережі).

Час реакції мережі зазвичай складається з декількох складових. У загальному випадку в нього входить час підготовки запитів на клієнтському комп'ютері, час передачі запитів між клієнтом і сервером через сегменти мережі й проміжне комунікаційне обладнання, час обробки запитів на сервері, час передачі відповідей клієнту від сервера та час обробки отримуваних від сервера відповідей на клієнтському комп'ютері.

Зрозуміло, що користувача не цікавить розкладання часу реакції на складові – йому важливий кінцевий результат, однак для мережного фахівця вельми важливо виділити із загального часу реакції складові, відповідні етапам власне мережної обробки даних, – передачу даних від клієнта до сервера через сегменти мережі й комунікаційне обладнання.

Знання мережних складових часу реакції дає можливість оцінити продуктивність окремих елементів мережі, виявити вузькі місця і, якщо буде потреба, виконати модернізацію мережі для підвищення її загальної продуктивності.

Пропускна здатність відбиває обсяг даних, переданих мережею або її частиною в одиницю часу. Пропускна здатність уже не є користувальницькою характеристикою, оскільки вона говорить про швидкість виконання внутрішніх операцій мережі – передачі пакетів даних між вузлами мережі через різні комунікаційні пристрої. Але вона безпосередньо характеризує якість виконання основної функції мережі – транспортування повідомлень – і тому частіше використовується під час аналізу продуктивності мережі, ніж час реакції.

Пропускна здатність вимірюється або в бітах у секунду, або в пакетах у секунду. Пропускна здатність може бути миттєвою, максимальною й середньою.

Середня пропускна здатність обчислюється шляхом ділення загального обсягу переданих даних на час їхньої передачі, причому вибирається досить тривалий проміжок часу – година, день або тиждень.

Миттєва пропускна здатність відрізняється від середньої тим, що для усереднення обирається дуже маленький проміжок часу, – наприклад, 10 мс або 1 с.

Максимальна пропускна здатність – це найбільша миттєва пропускна здатність, зафіксована протягом періоду спостереження.

Найчастіше під час проектування, настроювання та оптимізації мережі використовуються такі показники, як середня й максимальна пропускні здатності.

Середня пропускна здатність окремого елемента або всієї мережі дозволяє оцінити роботу мережі на великому проміжку часу, протягом якого в силу закону великих чисел піки та спади інтенсивності трафіка компенсують

один одного. Максимальна пропускна здатність дозволяє оцінити можливості мережі справлятися з піковими навантаженнями, характерними для особливих періодів роботи мережі, наприклад ранкових годин, коли співробітники підприємства майже водночас реєструються в мережі й звертаються до поділюваних файлів і баз даних.

Пропускню здатність можна вимірювати між будь-якими двома вузлами або точками мережі, наприклад, між клієнтським комп'ютером і сервером, між вхідним і вихідним портами маршрутизатора. Для аналізу й налаштування мережі дуже корисно знати дані про пропускну здатність окремих елементів мережі.

Важливо зазначити, що через послідовний характер передачі пакетів різними елементами мережі загальна пропускна здатність мережі будь-якого складеного шляху в мережі дорівнюватиме мінімальній із пропускних здатностей складових елементів маршруту. Для підвищення пропускної здатності складеного шляху необхідно насамперед звернути увагу на самі повільні елементи – у цьому випадку таким елементом, швидше за все, буде маршрутизатор. Слід підкреслити, що якщо переданий складеним шляхом трафік матиме середню інтенсивність, що перевершує середню пропускну здатність самого повільного елемента шляху, то черга пакетів до цього елемента буде рости теоретично нескінченно, а практично – доти, доки не заповниться його буферна пам'ять, а потім пакети просто почнуть відкидатися й губитися.

Іноді корисно оперувати із загальною пропускною здатністю мережі, що визначається як середня кількість інформації, переданої між всіма вузлами мережі в одиницю часу. Цей показник характеризує якість мережі в цілому, не диференціюючи його по окремих сегментах або пристроям.

Зазвичай при визначенні пропускної здатності сегмента або пристрою в переданих даних не виділяються пакети якогось певного користувача, додатка або комп'ютера – підраховується загальний обсяг переданої інформації. Проте, для більш точної оцінки якості обслуговування така деталізація бажана, і останнім часом системи управління мережами всі частіше дозволяють її виконувати.

Затримка передачі визначається як затримка між моментом надходження пакета на вхід будь-якого мережного пристрою або частини мережі й моментом появи його на виході цього пристрою. Цей параметр продуктивності за змістом близький реакції мережі, але відрізняється тим, що завжди характеризує лише мережні етапи обробки даних, без затримок

обробки комп'ютерами мережі. Звичайну якість мережі характеризують величинами максимальної затримки передачі й варіацією затримки. Не всі типи трафіка чутливі до затримок передачі, в усякому разі, до тих величин затримок, що характерні для комп'ютерних мереж, – зазвичай затримки не перевищують сотень мілісекунд, рідше – декількох секунд. Такого порядку затримки пакетів, породжувані файловими службами, службою електронної пошти або службою друку, мало впливають на якість цих служб з точки зору користувача мережі. З іншого боку, такі самі затримки пакетів, що переносять голосові дані або відеозображення, можуть приводити до значного зниження якості надаваної користувачу інформації – виникненню ефекту «луни», неможливості розібрати деякі слова, тремтіння зображення тощо.

Пропускна здатність і затримки передачі є незалежними параметрами, так що мережа може мати, наприклад, високу пропускну здатність, але вносити значні затримки при передачі кожного пакета. Приклад такої ситуації дає канал зв'язку, утворений геостаціонарним супутником. Пропускна здатність цього каналу може бути досить високою, наприклад, 2 Мбіт/с, у той час як затримка передачі завжди становить не менше 0,24 с, що визначається швидкістю поширення сигналу (близько 300000 км/с) і довжиною каналу (72000 км).

Надійність і безпека

Однією з початкових цілей створення розподілених систем, до яких належать і обчислювальні мережі, було досягнення більшої надійності порівняно з окремими обчислювальними машинами.

Важливо розрізняти кілька аспектів надійності. Для технічних пристроїв використовуються такі показники надійності, як середній час напрацювання на відмову, імовірність відмови, інтенсивність відмов. Але ці показники придатні для оцінки надійності простих елементів і пристроїв, які можуть перебувати лише у двох станах – працездатному або непрацездатному. Складні системи, що складаються з багатьох елементів, крім станів працездатності й непрацездатності, можуть мати й інші проміжні стани, які ці характеристики не враховують. У зв'язку з цим для оцінки надійності складних систем застосовується інший набір характеристик.

Готовність або *коефіцієнт готовності* (availability) означає проміжок часу, протягом якого система може бути використана. Готовність може бути покращена шляхом введення надмірності в структуру системи: ключові елементи системи мають існувати в декількох екземплярах, щоб під час відмови одного з них функціонування системи забезпечували інші.

Щоб систему можна було віднести до високонадійних, вона повинна, як мінімум, мати високу готовність, але цього недостатньо. Необхідно забезпечити збереження даних і захист їх від перекручувань. Крім того, має підтримуватися узгодженість (несуперечність) даних, наприклад, якщо для підвищення надійності на декількох файлових серверах зберігається кілька копій даних, то слід постійно забезпечувати їхню ідентичність.

Оскільки мережа працює на основі механізму передачі пакетів між кінцевими вузлами, то однією з характерних характеристик надійності є ймовірність доставки пакета вузлу призначення без перекручувань. Поряд із цією характеристикою можуть використовуватися й інші показники: імовірність втрати пакета (за будь-якої причини – через переповнення буфера маршрутизатора, через розбіжність контрольної суми, через відсутність працездатного шляху до вузла призначення тощо), імовірність перекручування окремого біта переданих даних, відношення втрачених пакетів до доставленого.

Іншим аспектом загальної надійності є *безпека* (security), тобто здатність системи захистити дані від несанкціонованого доступу. У розподіленій системі це зробити набагато складніше, ніж у централізованій. У мережах повідомлення передаються по лініях зв'язку, які часто проходять через загальнодоступні приміщення, де можуть бути встановлені засоби прослуховування ліній. Іншим уразливим місцем можуть бути залишені без догляду персональні комп'ютери. Крім того, завжди є потенційна загроза злому захисту мережі від неавторизованих користувачів, якщо мережа має виходи в глобальні мережі загального користування.

Ще однією характеристикою надійності є *відмовостійкість* (fault tolerance). У мережах під відмовостійкістю розуміється здатність системи сховати від користувача відмову окремих її елементів. Наприклад, якщо копії таблиці бази даних зберігаються водночас на декількох файлових серверах, то користувачі можуть просто не помітити відмову одного з них. У відмовостійкій системі відмова одного з її елементів приводить до деякого зниження якості її роботи (деградації), а не до повного останову. Так, при відмові одного з файлових серверів у попередньому прикладі збільшується тільки час доступу до бази даних через зменшення ступеня розпаралелювання запитів, але в цілому система буде продовжувати виконувати свої функції.

Розширюваність і масштабованість

Терміни *розширюваність* і *масштабованість* іноді використовують як синоніми, але це неправильно – кожен з них має чітко визначене самостійне значення.

Розширюваність (extensibility) означає можливість порівняно легкого додавання окремих елементів мережі (користувачів, комп'ютерів, додатків, служб), нарощування довжини сегментів мережі й заміни існуючої апаратури більш потужною. У цьому разі принципово важливо, що легкість розширення системи іноді може забезпечуватися в досить обмежених межах. Наприклад, локальна мережа Ethernet, побудована на основі одного сегмента товстого коаксіального кабелю, має гарну розширюваність, у тому розумінні, що дозволяє легко підключати нові станції. Однак така мережа має обмеження на кількість станцій, – їхнє число не має перевищувати 30–40. Хоча мережа допускає фізичне підключення до сегмента й більшої кількості станцій (до 100), але у цьому випадку найчастіше різко знижується продуктивність мережі. Наявність такого обмеження і є ознакою поганої масштабованості системи при гарній розширюваності.

Масштабованість (scalability) означає, що мережа дозволяє нарощувати кількість вузлів і довжину зв'язків у дуже широких межах, при цьому продуктивність мережі не погіршується. Для забезпечення масштабованості мережі доводиться застосовувати додаткове комунікаційне обладнання та спеціальним чином структурувати мережу. Наприклад, гарною масштабованістю володіє багатосегментна мережа, побудована з використанням комутаторів і маршрутизаторів і має ієрархічну структуру зв'язків. Така мережа може включати кілька тисяч комп'ютерів і при цьому забезпечувати кожному користувачу мережі потрібну якість обслуговування.

Прозорість

Прозорість (transparency) мережі досягається в тому випадку, коли мережа представляється користувачам не як множина окремих комп'ютерів, зв'язаних між собою складною системою кабелів, а як єдина традиційна обчислювальна машина з системою поділу часу. Відоме гасло компанії Sun Microsystems: «Мережа – це комп'ютер» – говорить саме про таку прозору мережу.

Прозорість може бути досягнута на двох різних рівнях – на рівні користувача й на рівні програміста. На рівні користувача прозорість означає, що для роботи з віддаленими ресурсами він використовує ті самі команди й звичні йому процедури, що й для роботи з локальними ресурсами. На програмному рівні прозорість полягає в тому, що додатку для доступу до віддалених ресурсів потрібні ті самі виклики, що й для доступу до локальних ресурсів. Прозорість на рівні користувача досягається простіше, оскільки всі

особливості процедур, пов'язані з розподіленим характером системи, маскуються від користувача програмістом, який створює додаток. Прозорість на рівні додатка вимагає приховування всіх деталей розподіленості засобами мережної операційної системи.

Мережа має приховувати всі особливості операційних систем і розходження в типах комп'ютерів. Користувач комп'ютера Macintosh повинен мати можливість звертатися до ресурсів, які підтримуються UNIX-системою, а користувач UNIX повинен мати можливість розділяти інформацію з користувачами Windows 95. Переважна кількість користувачів нічого не хоче знати про внутрішні формати файлів або про синтаксис команд UNIX. Користувач термінала IBM 3270 повинен мати можливості обмінюватися повідомленнями з користувачами мережі персональних комп'ютерів без необхідності вникати в секрети адрес, які запам'ятовуються важко.

Концепція прозорості може бути застосована до різних аспектів мережі. Наприклад, прозорість розташування означає, що від користувача не потрібно знань про місце розташування програмних і апаратних ресурсів, таких як процесори, принтери, файли й бази даних. Ім'я ресурсу не має включати інформацію про місце його розташування, тому імена типу `machine1:prog.c` або `\\ftp_serv\\pub` прозорими не є. Аналогічно, прозорість переміщення означає, що ресурси мають вільно переміщатися з одного комп'ютера в інший без зміни своїх імен. Ще одним з можливих аспектів прозорості є прозорість паралелізму, яка полягає в тому, що процес розпаралелювання обчислень відбувається автоматично, без участі програміста, при цьому система сама розподіляє паралельні галузі додатка по процесорах і комп'ютерах мережі. На сьогодні не можна сказати, що прозорість повною мірою властива багатьом обчислювальним мережам, це, скоріше, мета, до якої прагнуть розробники сучасних мереж.

Підтримка різних видів трафіка

Комп'ютерні мережі від самого початку призначені для спільного доступу користувача до ресурсів комп'ютерів: файлам, принтерам тощо. Трафік, що створюється цими традиційними службами комп'ютерних мереж, має свої особливості й істотно відрізняється від трафіка повідомлень у телефонних мережах або, наприклад, у мережах кабельного телебачення. Однак 90-і роки стали роками проникнення в комп'ютерні мережі трафіка мультимедійних даних, що подають у цифровій формі мову та відеозображення. Комп'ютерні мережі стали використовуватися для організації відеоконференцій, навчання й розваг на основі відеофільмів тощо. Природно,

що для динамічної передачі мультимедійного трафіка потрібні інші алгоритми та протоколи й, відповідно, інше обладнання. Хоча частка мультимедійного трафіка поки невелика, він уже почав своє проникнення як у глобальні, так і локальні мережі, і цей процес, мабуть, буде тривати зі зростаючою швидкістю.

Головною особливістю трафіка, що утворюється при динамічній передачі голосу або зображення, є наявність жорстких вимог до синхронності повідомлень. Для якісного відтворення безперервних процесів, якими є звукові коливання або зміни інтенсивності світла у відеозображенні, необхідно отримання обмірюваних і закодованих амплітуд сигналів з тією самою частотою, з якою вони обмірювалися на передавальній стороні. При запізнюванні повідомлень спостерігатимуться перекручування.

Водночас трафік комп'ютерних даних характеризується вкрай нерівномірною інтенсивністю надходження повідомлень у мережу за відсутності жорстких вимог до синхронності доставки цих повідомлень. Наприклад, доступ користувача, що працює з текстом на віддаленому диску, породжує випадковий потік повідомлень між віддаленим і локальним комп'ютерами, що залежить від дій користувача з редагування тексту, причому затримки при доставці в певних (і досить широких з комп'ютерної точки зору) межах мало впливають на якість обслуговування користувача мережі. Всі алгоритми комп'ютерного зв'язку, відповідні протоколи та комунікаційне обладнання були розраховані саме на такий «пульсуючий» характер трафіка, тому необхідність передавати мультимедійний трафік вимагає внесення принципових змін як у протоколи, так і в обладнання. Сьогодні практично всі нові протоколи в тому або іншому ступені надають підтримку мультимедійного трафіка.

Особливу складність представляє поєднання в одній мережі традиційного комп'ютерного та мультимедійного трафіка. Передача винятково мультимедійного трафіка комп'ютерною мережею хоча й пов'язана з певними складностями, але викликає менші труднощі. А от випадок співіснування двох типів трафіка з протилежними вимогами до якості обслуговування є набагато складнішою задачею. Зазвичай протоколи й обладнання комп'ютерних мереж відносять мультимедійний трафік до факультативного, тому якість його обслуговування залишає бажати кращого. Сьогодні витрачаються великі зусилля на створення мереж, які не обмежують інтереси одного з типів трафіка. Найбільш близькі до цього мережі на основі технології АТМ, розробники якої від самого початку враховували випадок співіснування різних типів трафіка в одній мережі.

Керованість

Керованість мережі має на увазі можливість централізовано контролювати стан основних елементів мережі, виявляти й розв'язувати проблеми, які виникають в ході роботи мережі, виконувати аналіз продуктивності й планувати розвиток мережі. В ідеалі засоби управління мережами є системою, що здійснює спостереження, контроль і управління кожним елементом мережі – від найпростіших до найскладніших пристроїв, при цьому така система розглядає мережу як єдине ціле, а не як розрізнений набір окремих пристроїв.

Гарна система управління спостерігає за мережею й, виявивши проблему, активізує певну дію, виправляє ситуацію й повідомляє адміністратора про те, що сталося і які кроки зроблено. Водночас із цим система управління має накопичувати дані, на підставі яких можна планувати розвиток мережі. Нарешті, система управління має бути незалежною від виробника й мати зручний інтерфейс, що дозволяє виконувати всі дії з однієї консолі.

Вирішуючи тактичні задачі, адміністратори й технічний персонал стикаються зі щоденними проблемами забезпечення працездатності мережі. Ці задачі вимагають швидкого вирішення, обслуговуючий мережу персонал повинен оперативно реагувати на повідомлення про несправності, які надходять від користувачів або автоматичних засобів управління мережею. Поступово стають помітні більш загальні проблеми продуктивності, конфігурування мережі, обробки збоїв та безпеки даних, що вимагають стратегічного підходу, тобто планування мережі. Планування, крім того, включає прогноз змін вимог користувачів до мережі, питання застосування нових додатків, нових мережних технологій тощо.

Корисність системи управління особливо яскраво проявляється у великих мережах: корпоративних або публічних глобальних. Без системи управління в таких мережах потрібна присутність кваліфікованих фахівців з експлуатації в кожному будинку кожного міста, де встановлено обладнання мережі, що в підсумку призводить до необхідності утримання величезного штату обслуговуючого персоналу.

На сьогодні в галузі систем управління мережами багато невирішених проблем. Явно недостатньо дійсно зручних, компактних і багатопроTOCOLьних засобів управління мережею. Більшість існуючих засобів зовсім не управляють мережею, а всього лише здійснюють спостереження за її роботою. Вони стежать за мережею, але не виконують активних дій, якщо з мережею щось відбулося або може статися. Мало масштабованих систем, здатних

обслуговувати як мережі масштабу відділу, так і мережі масштабу підприємства, – дуже багато систем управляють тільки окремими елементами мережі й не аналізують здатність мережі виконувати якісну передачу даних між кінцевими користувачами мережі.

Сумісність

Сумісність або інтегрованість означає, що мережа здатна містити найрізноманітніше програмне та апаратне забезпечення, тобто в ній можуть співіснувати різні операційні системи, які підтримують різні стеки комунікаційних протоколів, і працювати апаратні засоби й додатки від різних виробників. Мережа, що складається з різнотипних елементів, називається неоднорідною або гетерогенною, а якщо гетерогенна мережа працює без проблем, то вона є інтегрованою. Основний шлях побудови інтегрованих мереж – використання модулів, виконаних відповідно до відкритих стандартів і специфікацій.

Висновки

1. Якість роботи мережі характеризують такі властивості: надійність, сумісність, керованість, захищеність, продуктивність, розширюваність і масштабованість.

2. Існують два основних підходи до забезпечення якості роботи мережі. *Перший полягає в тому*, що мережа гарантує користувачу дотримання деякої числової величини показника якості обслуговування. Наприклад, мережі frame relay і АТМ можуть гарантувати користувачу заданий рівень пропускну здатності. *При другому підході (best effort)* мережа намагається за можливістю більш якісно обслужити користувача, але нічого при цьому не гарантує.

3. До основних характеристик продуктивності мережі належать: час реакції, що визначається як час між виникненням запиту до будь-якого мережного сервісу й отриманням відповіді на нього; пропускна здатність, що відображає обсяг даних, переданих мережею в одиницю часу, і затримка передачі, що дорівнює інтервалу між моментом надходження пакета на вхід будь-якого мережного пристрою й моментом його появи на виході цього пристрою.

4. Для оцінки надійності мереж використовуються різні характеристики, такі як: коефіцієнт готовності, що означає відрізок часу, протягом якого система може бути використана; безпека, тобто здатність системи захистити дані від несанкціонованого доступу; відмовостійкість – здатність системи працювати в умовах відмови деяких її елементів.

5. Розширюваність означає можливість порівняно легкого додавання окремих елементів мережі (користувачів, комп'ютерів, додатків, сервісів), нарощування довжини сегментів мережі й заміни існуючої апаратури більш потужною.

6. Масштабованість означає, що мережа дозволяє нарощувати кількість вузлів і довжину зв'язків у дуже широких межах, при цьому продуктивність мережі не погіршується.

7. Прозорість – властивість мережі приховувати від користувача деталі свого внутрішнього устрою, спрощуючи тим самим його роботу в мережі.

8. Під керованістю мережі розуміють можливість централізовано контролювати стан основних елементів мережі, виявляти та розв'язувати проблеми, які виникають в ході роботи мережі, виконувати аналіз продуктивності й планувати розвиток мережі.

9. Сумісність означає, що мережа здатна містити в собі найрізноманітніше програмне й апаратне забезпечення.

3.3 Контрольні запитання та завдання

1. З наведеної нижче послідовності назв стандартних стеків комунікаційних протоколів виділіть назви, які належать до одного і того самого стеку: TCP/IP, Microsoft, IPX/SPX, Novell, Internet, Do, NetBIOS/SMB, DECnet.

2. Поясніть, у чому відмінність локальних мереж від глобальних на рівні служб? На рівні транспортної системи?

3. Назвіть найчастіше використовувані характеристики продуктивності мережі.

4. Поясніть, що важливіше для передачі мультимедійного трафіка: надійність чи синхронність?

5. Поясніть значення деяких мережних характеристик, назви яких знаходяться в англomовному написанні: availability, fault tolerance, security, extensibility, scalability, transparency.

Лекція 4

ХАРАКТЕРИСТИКА ФІЗИЧНОГО РІВНЯ АРХІТЕКТУРИ КОМП'ЮТЕРНИХ МЕРЕЖ

4.1 Типи ліній зв'язку

Лінія зв'язку (рис. 4.1) складається, в загальному випадку, з фізичного середовища, через яке передаються електричні інформаційні сигнали апаратури передачі даних і проміжної апаратури. Синонімом терміна «лінія зв'язку» (line) є термін «канал зв'язку» (channel).

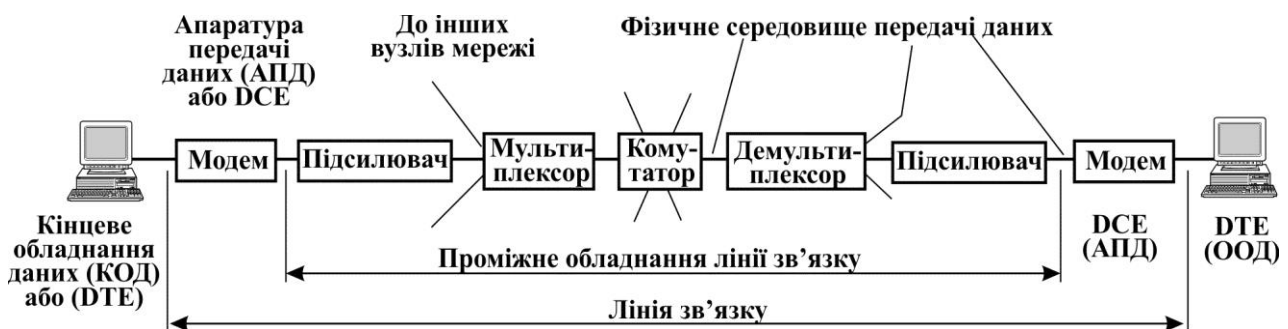


Рисунок 4.1 – Склад лінії зв'язку

Фізичне середовище передачі даних (medium) може бути кабелем, тобто набором проводів, ізоляційних і захисних оболонок і сполучних рознімів, а також земною атмосферою або космічним простором, через які поширюються електромагнітні хвилі.

Залежно від середовища передачі даних лінії зв'язку поділяються на такі (рис. 4.2):

- проводові (повітряні);
- кабельні (мідні й волоконно-оптичні);
- радіоканали наземного та супутникового зв'язку.

Проводові (повітряні) лінії зв'язку — це є проводи без будь-яких ізолюючих або екрануючих обплетень, прокладені між стовпами і висять у повітрі. По таких лініях зв'язку традиційно передаються телефонні або телеграфні сигнали, але за відсутності інших можливостей ці лінії використовуються також для передачі комп'ютерних даних. Швидкісні характеристики й завадозахищеність цих ліній залишають бажати кращого. На сьогодні проводові лінії зв'язку швидко витісняються кабельними.

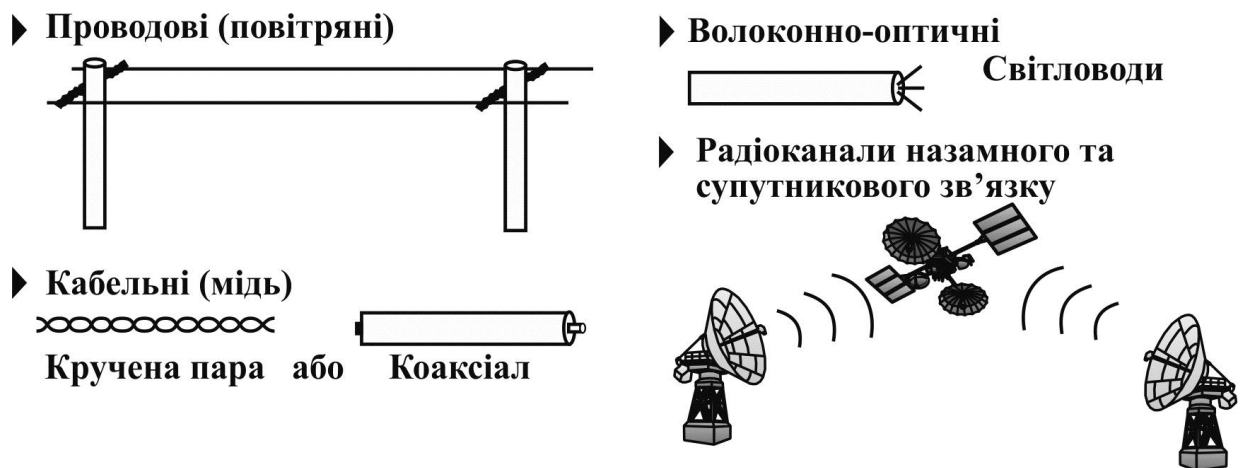


Рисунок 4.2 – Типи ліній зв'язку

Кабельні лінії є досить складною конструкцією. Кабель складається з провідників, укладених у кілька шарів ізоляції: електричної, електромагнітної, механічної, а також, можливо, кліматичної. Крім того, кабель може бути оснащений рознімами, які дозволяють швидко виконувати приєднання до нього різного обладнання. У комп'ютерних мережах застосовуються три основних типи кабелю: на основі скручених пар мідних проводів, коаксіальні – з мідною жилою, а також волоконно-оптичні.

Скручена пара проводів називається **крученою парою** (twisted pair). Кручена пара виконується в *екранованому варіанті* (Shielded Twisted Pair, STP), коли пари мідних проводів обертається в ізоляційний екран, і *неекранованому* (Unshielded Twisted Pair, UTP), коли ізоляційна обгортка відсутня. Скручування проводів зменшує вплив зовнішніх завад на корисні сигнали, передані по кабелю. **Коаксіальний кабель** (coaxial) має несиметричну конструкцію й складається з внутрішньої мідної жили й обплетення, відділеної від жили шаром ізоляції. Існує кілька типів коаксіального кабелю, які відрізняються характеристиками й галузями застосування – для локальних мереж, для глобальних мереж, для кабельного телебачення тощо. **Волоконно-оптичний кабель** (optical fiber) складається з тонких (5–60 мікрон) волокон, по яких поширюються світлові сигнали. Це найбільш якісний тип кабелю – він забезпечує передачу даних з дуже високою швидкістю (близько 10 Гбіт/с і вище) і до того ж краще інших типів передавального середовища забезпечує захист даних від зовнішніх завад.

Радіоканали наземного й супутникового зв'язку утворюються за допомогою передавача й приймача радіохвиль. Існує велика кількість різних типів радіоканалів, які розрізняються як використанням частотним діапазоном, так і дальністю каналу. Діапазони коротких, середніх і довгих хвиль (КВ, СВ і ДВ) називаються також діапазонами амплітудної модуляції

(Amplitude Modulation, AM) за типом використовуваного в них методу модуляції сигналу, забезпечують далекий зв'язок, але при невисокій швидкості передачі даних. Більш швидкісними є канали, що працюють на діапазонах ультракоротких хвиль (УКВ), для яких характерна частотна модуляція (Frequency Modulation, FM), а також діапазонах надвисоких частот (СВЧ або microwaves). У діапазоні СВЧ (понад 4 ГГц) сигнали вже не відбиваються іоносферою Землі й для стійкого зв'язку потрібна наявність прямої видимості між передавачем та приймачем. Тому такі частоти використовують або супутникові канали, або радіорелейні канали, де ця умова виконується.

У комп'ютерних мережах сьогодні застосовуються практично всі описані типи фізичних середовищ передачі даних, але найперспективнішими є волоконно-оптичні. На них сьогодні будуються як магістралі великих територіальних мереж, так і високошвидкісні лінії зв'язку локальних мереж. Популярним середовищем є також кручена пара, яка характеризується відмінним співвідношенням якості до вартості, а також простотою монтажу. За допомогою крученої пари зазвичай підключають кінцевих абонентів мереж на відстанях до 100 метрів від концентратора. Супутникові канали й радіозв'язок використовуються найчастіше в тих випадках, коли кабельні зв'язки застосовувати не можна – наприклад, якщо канал проходить через малонаселену місцевість або ж для зв'язку з мобільним користувачем мережі, таким як шофер вантажівки, лікар, що робить обхід тощо.

Апаратура ліній зв'язку

Апаратура передачі даних (АПД або DCE – Data Circuit terminating Equipment) безпосередньо зв'язує комп'ютери або локальні мережі користувача з лінією зв'язку і, отже, є, приміжовим обладнанням. Традиційно апаратуру передачі даних включають до складу лінії зв'язку. Прикладами ОСІ є модеми, термінальні адаптери мереж ISDN, оптичні модеми, пристрої підключення до цифрових каналів. Зазвичай DCE працює на фізичному рівні, відповідаючи за передачу та прийняття сигналу потрібної форми й потужності у фізичне середовище.

Апаратура користувача лінії зв'язку, яка виробляє дані для передачі по лінії зв'язку, і яка підключається безпосередньо до апаратури передачі даних, узагальнено називається кінцевим обладнанням даних (КОД або DTE – Data Terminal Equipment). Прикладом DTE можуть слугувати комп'ютери або маршрутизатори локальних мереж. Цю апаратуру не включають до складу лінії зв'язку.

Поділ обладнання на класи DCE і DTE у локальних мережах є досить умовним. Наприклад, адаптер локальної мережі можна вважати як приладом комп'ютера, тобто DTE, так і складовою частиною каналу зв'язку, тобто DCE.

Проміжна апаратура зазвичай використовується на лініях зв'язку великої довжини і вирішує дві основні задачі:

- покращення якості сигналу;
- створення постійного складеного каналу зв'язку між двома абонентами мережі.

У локальних мережах проміжна апаратура може зовсім не використовуватися, якщо довжина фізичного середовища – кабелів або радіоефіру – дозволяє одному мережному адаптеру приймати сигнали безпосередньо від іншого мережного адаптера, без додаткового посилення. У протилежному випадку застосовуються пристрої типу повторювачів і концентраторів.

У глобальних мережах необхідно забезпечити якісну передачу сигналів на відстані в сотні й тисячі кілометрів. Тому без підсилювачів сигналів, установлених через певні відстані, побудувати територіальну лінію зв'язку неможливо. У глобальній мережі необхідна також і проміжна апаратура іншого роду – мультиплексори, демультимплексори та комутатори. Ця апаратура вирішує другу зазначену задачу, тобто створює між двома абонентами мережі складений канал з відрізків фізичного середовища, які не комутуються – кабелів з підсилювачами. Важливо зазначити, що наведені на рис. 2.1 мультиплексори, демультимплексори та комутатори утворюють складений канал на довгостроковій основі, наприклад, на місяць або рік, причому абонент не може впливати на процес комутації цього каналу, – ці пристрої керуються з окремих входів, які абоненту недоступні (на рисунку не показані). Наявність проміжної комутаційної апаратури позбавляє творців глобальної мережі від необхідності прокладати окрему кабельну лінію для кожної пари вузлів мережі, які з'єднуються. Замість цього між мультиплексорами й комутаторами використовується високошвидкісне фізичне середовище, наприклад, волоконно-оптичний або коаксіальний кабель, по якому передаються водночас дані від великої кількості порівняно низькошвидкісних абонентських ліній. А коли потрібно утворити постійне з'єднання між будь-якими двома кінцевими вузлами мережі, що знаходяться, наприклад, у різних містах, то мультиплексори, комутатори та демультимплексори налаштовуються оператором каналу відповідно до умов. Високошвидкісний канал зазвичай називають ущільненим каналом.

Проміжна апаратура каналу зв'язку прозора для користувача, він її не помічає й не враховує у своїй роботі. Для нього важливі тільки якість отриманого каналу, яка впливає на швидкість передачі дискретних даних. Насправді ж проміжна апаратура утворює складну мережу, яку називають первинною мережею, оскільки сама по собі вона ніяких високорівневих служб (наприклад, файлової або передачі голосу) не підтримує, а тільки є основою для побудови комп'ютерних, телефонних або інших мереж.

Залежно від типу проміжної апаратури всі лінії зв'язку поділяються на аналогові й цифрові. В аналогових лініях проміжна апаратура призначена для посилення аналогових сигналів, тобто сигналів, які мають безперервний діапазон значень. Такі лінії зв'язку традиційно застосовувалися в телефонних мережах для зв'язку АТС між собою. Для створення високошвидкісних каналів, які мультиплексують кілька низькошвидкісних аналогових абонентських каналів, при аналоговому підході зазвичай використовується техніка частотного мультиплексування (Frequency Division Multiplexing, FDM).

У цифрових лініях зв'язку передані сигнали мають кінцеву кількість станів. Як правило, елементарний сигнал, тобто сигнал, переданий за один такт роботи передавальної апаратури, має 2 або 3 стана, які передаються в лініях зв'язку імпульсами прямокутної форми. За допомогою таких сигналів передаються як комп'ютерні дані, так і оцифровані мова та зображення. У цифрових каналах зв'язку використовується проміжна апаратура, яка покращує форму імпульсів і забезпечує їх ресинхронізацію, тобто відновлює період їхнього проходження. Проміжна апаратура утворення високошвидкісних цифрових каналів (мультиплексори, демультиплексори, комутатори) працює за принципом часового мультиплексування каналів (Time Division Multiplexing, TDM), коли кожному низькошвидкісному каналу виділяється певна частка часу (тайм-слот або квант) високошвидкісного каналу.

Апаратура передачі дискретних комп'ютерних даних по аналогових і цифрових лініях зв'язку істотно відрізняється, оскільки в першому випадку лінія зв'язку призначена для передачі сигналів довільної форми й не пред'являє ніяких вимог до способу подання одиниць і нулів апаратурою передачі даних, а в другому – всі параметри переданих лінією імпульсів стандартизовані. Іншими словами, на цифрових лініях зв'язку протокол фізичного рівня визначений, а на аналогових лініях – ні.

4.2 Характеристики ліній зв'язку

Типи характеристик і способи їхнього визначення

До основних характеристик ліній зв'язку належать:

- амплітудно-частотна характеристика;
- смуга пропуску;
- загасання;
- завадостійкість;
- перехресні наведення на ближньому кінці лінії;
- пропускна здатність;
- вірогідність передачі даних;
- питома вартість.

Перш за все розробника обчислювальної мережі цікавлять пропускна здатність і вірогідність передачі даних, оскільки ці характеристики прямо впливають на продуктивність і надійність створюваної мережі. Пропускна здатність і вірогідність – це характеристики як лінії зв'язку, так і способу передачі даних. Тому, якщо спосіб передачі (протокол) уже визначений, то відомі й ці характеристики. Наприклад, пропускна здатність цифрової лінії завжди відома, оскільки на ній визначений протокол фізичного рівня, який задає бітову швидкість передачі даних – 64 Кбіт/с, 2 Мбіт/с тощо.

Однак не можна говорити про пропускну здатність лінії зв'язку до того, як для неї буде визначений протокол фізичного рівня. Саме в таких випадках, коли тільки належить визначити, який з безлічі існуючих протоколів можна використовувати на даній лінії, дуже важливими є інші характеристики лінії, такі як смуга пропускання, перехресні наведення, завадостійкість та ін.

Для визначення характеристик лінії зв'язку часто використовують аналіз її реакцій на деякі еталонні впливи. Такий підхід дозволяє досить просто й однотипно визначати характеристики ліній зв'язку будь-якої природи, не вдаючись до складних теоретичних досліджень. Найчастіше як еталонні сигнали для дослідження реакцій ліній зв'язку використовуються синусоїдальні сигнали різних частот. Це пов'язано з тим, що сигнали цього типу часто зустрічаються в техніці і з їхньою допомогою можна подати будь-яку функцію часу – як безперервний процес коливальних звуку, так і прямокутні імпульси, що генеруються комп'ютером.

Спектральний аналіз сигналів на лініях зв'язку

З теорії гармонійного аналізу відомо, що будь-який періодичний процес можна подати у вигляді суми синусоїдальних коливань різних частот і різних амплітуд (рис. 4.3). Кожна складова синусоїда називається також гармонікою, а набір всіх гармонік називають спектральним розкладанням вихідного сигналу. Неперіодичні сигнали можна подати у вигляді інтеграла синусоїдальних сигналів з безперервним спектром частот.

Наприклад, спектральне розкладення ідеального імпульсу (одиничної потужності й нульової тривалості) має складові всього спектра частот, від $-\infty$ до $+\infty$ (рис. 4.4).

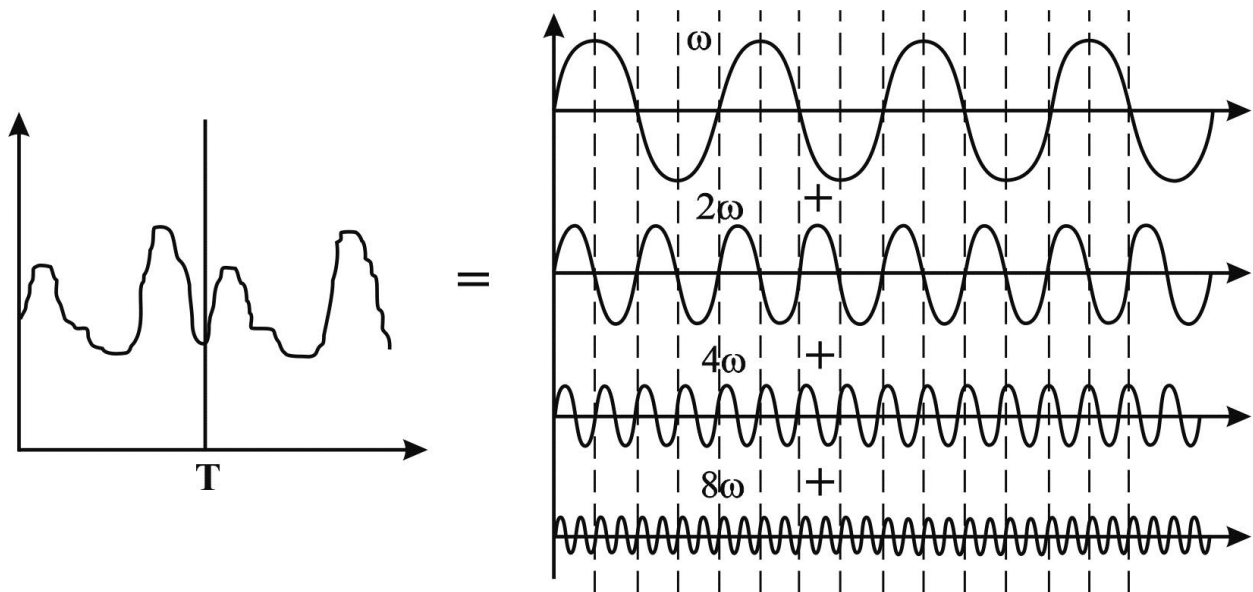


Рисунок 4.3 – Подання періодичного сигналу сумою синусоїд

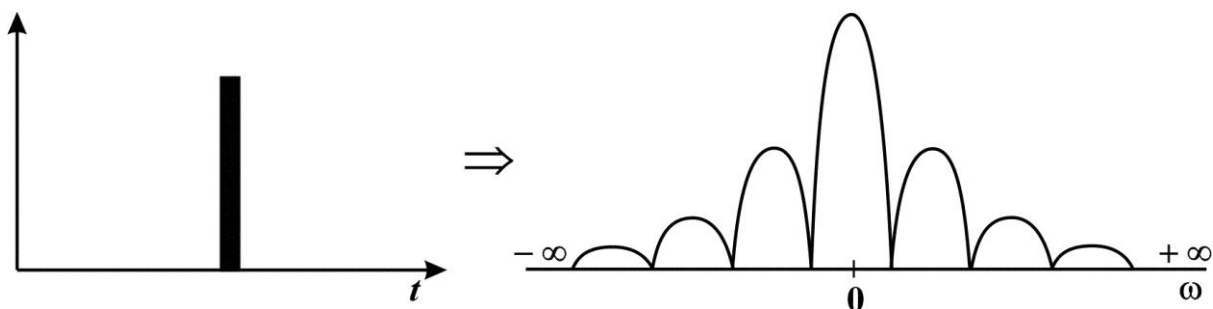


Рисунок 4.4 – Спектральне розкладення ідеального імпульсу

Техніка знаходження спектра будь-якого вихідного сигналу добре відома. Для деяких сигналів, які добре описуються аналітично (наприклад, для послідовності прямокутних імпульсів однакової тривалості й амплітуди), спектр легко обчислюється на базі формул Фур'є. Для сигналів довільної

форми, що зустрічаються на практиці, спектр можна знайти за допомогою спеціальних приладів – спектральних аналізаторів, які вимірюють спектр реального сигналу й відображають амплітуди складових гармонік на екрані або роздруковують їх на принтері.

Перекручування передавальним каналом синусоїди будь-якої частоти призводить в остаточному підсумку до перекручування переданого сигналу будь-якої форми, особливо якщо синусоїди різних частот спотворюються неоднаково. Якщо це аналоговий сигнал, який передає мову, то змінюється тембр голосу за рахунок перекручування обертонів – бічних частот. При передачі імпульсних сигналів, характерних для комп'ютерних мереж, спотворюються низькочастотні й високочастотні гармоніки, в результаті фронти імпульсів втрачають свою прямокутну форму (рис. 4.5). Внаслідок цього на приймальному кінці лінії сигнали можуть погано розпізнаватися.



Рисунок 4.5 – Перекручування імпульсів у лінії зв'язку

Лінія зв'язку перекручує передані сигнали через те, що її фізичні параметри відрізняються від ідеальних. Так, наприклад, мідний провід завжди є деякою розподіленою по довжині комбінацією активного опору, ємнісного й індуктивного навантаження (рис. 4.6). У результаті для синусоїд різних частот лінія матиме різний повний опір, а отже, і передаватимуться вони неоднаково. Волоконно-оптичний кабель також має відхилення, які заважають ідеальному поширенню світла. Якщо лінія зв'язку містить проміжну

апаратуру, то вона також може вносити додаткові перекошування, оскільки неможливо створити пристрій, який би однаково добре передавав увесь спектр синусоїд, від нуля до нескінченності.

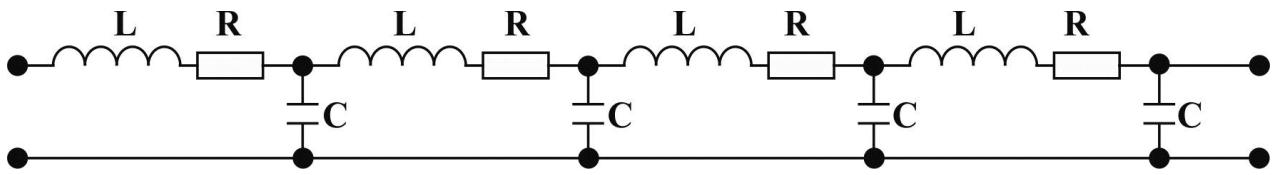


Рисунок 4.6 – Подання лінії як розподіленого індуктивно-ємнісного навантаження

Крім перекошувань сигналів, внесених внутрішніми фізичними параметрами лінії зв'язку, існують і зовнішні завади, які роблять свій внесок у перекошування форми сигналів на виході лінії. Ці завади створюють різні електричні двигуни, електронні пристрої, атмосферні явища тощо. Незважаючи на захисні заходи, що застосовуються розробниками кабелів і підсилювально-комутуючої апаратури, повністю компенсувати вплив зовнішніх завад не вдається. Тому сигнали на виході лінії зв'язку зазвичай мають складну форму (як це й показано на рис. 4.5), по якій іноді важко зрозуміти, яка дискретна інформація була подана на вхід лінії.

Амплітудно-частотна характеристика, смуга пропускання й загасання

Ступінь перекошування синусоїдальних сигналів лініями зв'язку оцінюється за допомогою таких характеристик, як амплітудно-частотна характеристика, смуга пропускання й загасання на певній частоті.

Амплітудно-частотна характеристика (рис. 4.7) показує, як загасає амплітуда синусоїди на виході лінії зв'язку порівняно з амплітудою на її вході для всіх можливих частот переданого сигналу. Замість амплітуди в цій характеристиці часто використовують також такий параметр сигналу, як його потужність.

Знання амплітудно-частотної характеристики реальної лінії дозволяє визначити форму вихідного сигналу практично для будь-якого вхідного сигналу. Для цього необхідно знайти спектр вхідного сигналу, перетворити амплітуду гармонік, які його складають, відповідно до амплітудно-частотної характеристики, а потім знайти форму вихідного сигналу, склавши перетворені гармоніки.

Незважаючи на повноту інформації, що надається амплітудно-частотною характеристикою про лінію зв'язку, її використання ускладнюється тією обставиною, що отримати її досить важко. Адже для цього потрібно провести

тестування лінії еталонними синусоїдами на всьому діапазоні частот від нуля до деякого максимального значення, яке може зустрітися у вхідних сигналах. Причому змінювати частоту вхідних синусоїд потрібно з невеликим кроком, і виходить, що кількість експериментів має бути дуже великою. Тому на практиці замість амплітудно-частотної характеристики застосовуються інші, спрощені характеристики – смугу пропускання й загасання.

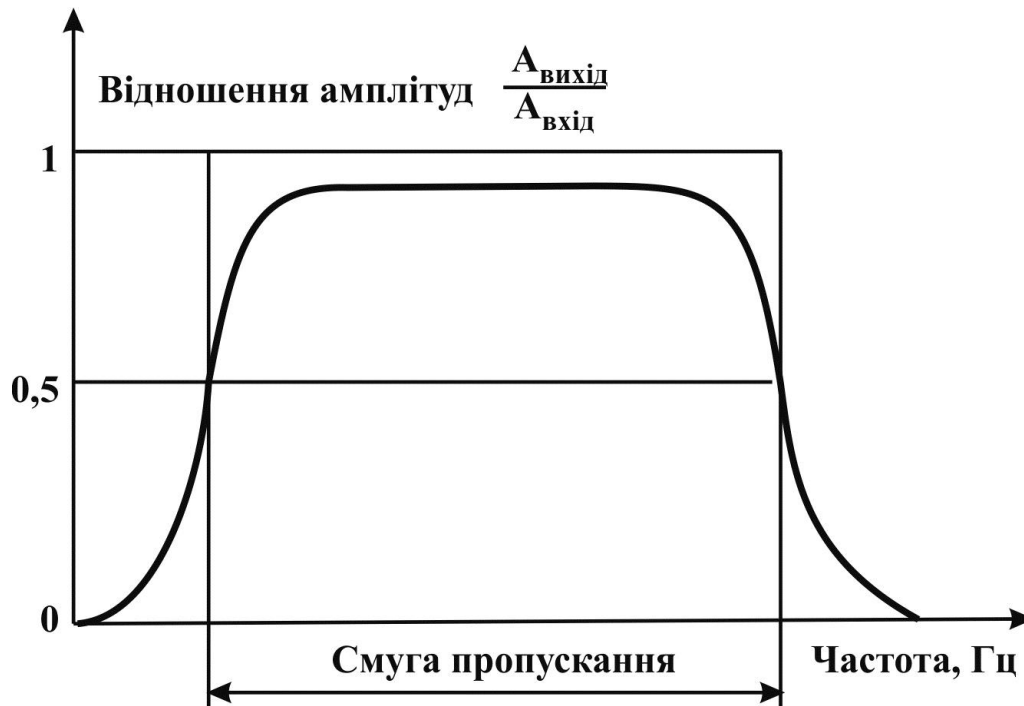


Рисунок 4.7 – Амплітудно-частотна характеристика

Смуга пропускання (bandwidth) – це безперервний діапазон частот, для якого відношення амплітуди вихідного сигналу до вхідного перевищує деяку заздалегідь задану межу, зазвичай 0,5. Тобто смуга пропускання визначає діапазон частот синусоїдального сигналу, за яких цей сигнал передається лінією зв'язку без значних перекручувань. Знання смуги пропускання дозволяє отримати з деяким ступенем наближення той самий результат, що й знання амплітудно-частотної характеристики. Як ми побачимо далі, ширина смуги пропускання найбільшою мірою впливає на максимально можливу швидкість передачі інформації лінією зв'язку. Саме цей факт знайшов відображення в англійському еквіваленті розглянутого терміна (width – ширина).

Загасання (attenuation) визначається як відносне зменшення амплітуди або потужності сигналу під час передачі по лінії сигналу певної частоти. Отже, загасання є однією точкою з амплітудно-частотної характеристики лінії. Часто під час експлуатації лінії заздалегідь відома основна частота переданого сигналу, тобто та частота, гармоніка якої має найбільшу амплітуду

й потужність. Тому достатньо знати загасання на цій частоті, щоб приблизно оцінити перекручування переданих лінією сигналів. Точніші оцінки можливі при знанні загасання на декількох частотах, що відповідають декільком основним гармонікам переданого сигналу.

Загасання A зазвичай вимірюється в децибелах (дб, decibel – d) і обчислюється за такою формулою:

$$A = 10 \log_{10} P_{\text{вих}} / P_{\text{вх}},$$

де $P_{\text{вих}}$ – потужність сигналу на виході лінії,

$P_{\text{вх}}$ – потужність сигналу на вході лінії.

Оскільки потужність вихідного сигналу кабелю без проміжних підсилювачів завжди менше, ніж потужність вхідного сигналу, загасання кабелю завжди є негативною величиною.

Наприклад, кабель на кручений парі категорії 5 характеризується загасанням не нижче –23,6 дб для частоти 100 МГц при довжині кабелю 100 м. Частота 100 МГц обрана тому, що кабель цієї категорії призначений для високошвидкісної передачі даних, сигнали яких мають значимі гармоніки із частотою приблизно 100 МГц. Кабель категорії 3 призначений для низькошвидкісної передачі даних, тому для нього визначається загасання на частоті 10 МГц (не нижче –11,5 дб). Часто оперують з абсолютними значеннями загасання, без зазначення знака.

Абсолютний рівень потужності, наприклад, рівень потужності передавача, також вимірюється в децибелах. При цьому як базове значення потужності сигналу, щодо якого вимірюється поточна потужність, приймається значення в 1 мВт. Отже, рівень потужності p обчислюється за формулою:

$$p = \frac{10 \log_{10} P}{1 \text{ мВт}} \text{ [дБм]},$$

де P – потужність сигналу в міліватах,

дБм (dBm) – одиниця виміру рівня потужності (децибел на 1 мВт).

Отже, амплітудно-частотна характеристика, смуга пропускання й загасання є універсальними характеристиками, і їхнє знання дозволяє зробити висновок про те, як через лінію зв'язку передаватимуться сигнали будь-якої форми.

Смуга пропускання залежить від типу лінії і її довжини. На рис. 4.8 показані смуги пропускання ліній зв'язку різних типів, а також найчастіше використовувані в техніці зв'язку частотні діапазони.

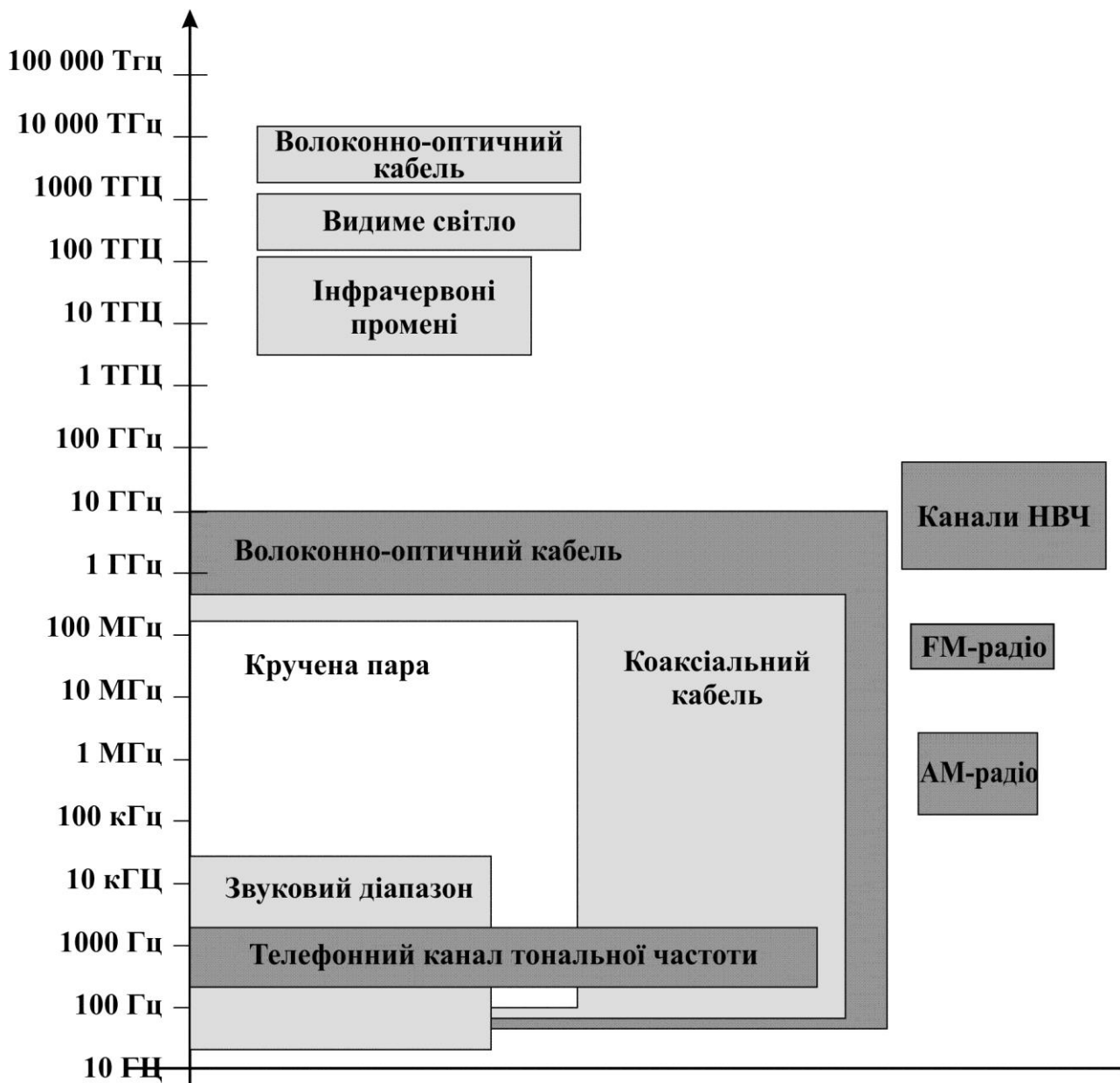


Рисунок 4.8 – Смуги пропускання ліній зв'язку й популярних частотних діапазонів

Пропускна здатність лінії

Пропускна здатність (throughput) лінії характеризує максимально можливу швидкість передачі даних по лінії зв'язку. Пропускна здатність вимірюється

в бітах у секунду – біт/с, а також у похідних одиницях, таких як кілобіт у секунду (Кбіт/с), мегабіт у секунду (Мбіт/с), гігабіт у секунду (Гбіт/с) тощо.

Пропускна здатність лінії зв'язку залежить не тільки від її характеристик, таких як амплітудно-частотна характеристика, але й від спектра переданих сигналів. Якщо значимі гармоніки сигнала (тобто ті гармоніки, амплітуди яких роблять основний внесок у результуючий сигнал) попадають у смугу

пропускання лінії, то такий сигнал добре передаватиметься даною лінією зв'язку, і приймач зможе правильно розпізнати інформацію, відправлену по лінії передавачем (рис. 4.9, а).

Вибір способу представлення дискретної інформації у вигляді сигналів, що подаються на лінію зв'язку, називається фізичним або лінійним кодуванням. Від обраного способу кодування залежить спектр сигналів і, відповідно, пропускна здатність лінії. Отже, для одного способу кодування лінія може мати одну пропускну здатність, а для іншого – іншу. Наприклад, кручена пара категорії 3 може передавати дані з пропускну здатністю 10 Мбіт/с за способом кодування стандарту фізичного рівня 10Base-T і 33 Мбіт/с за способом кодування стандарту 100Base-T4. У прикладі, наведеному на рис. 4.9, прийнятий такий спосіб кодування – логічна 1 подана на лінії позитивним потенціалом, а логічний 0 – негативним.

Якщо ж значущі гармоніки виходять за межі смуги пропускання лінії зв'язку, то сигнал значно спотворюватиметься, приймач помилятиметься під час розпізнавання інформації, а виходить, інформація не зможе передаватися із заданою пропускну здатністю (рис. 4.9, б).

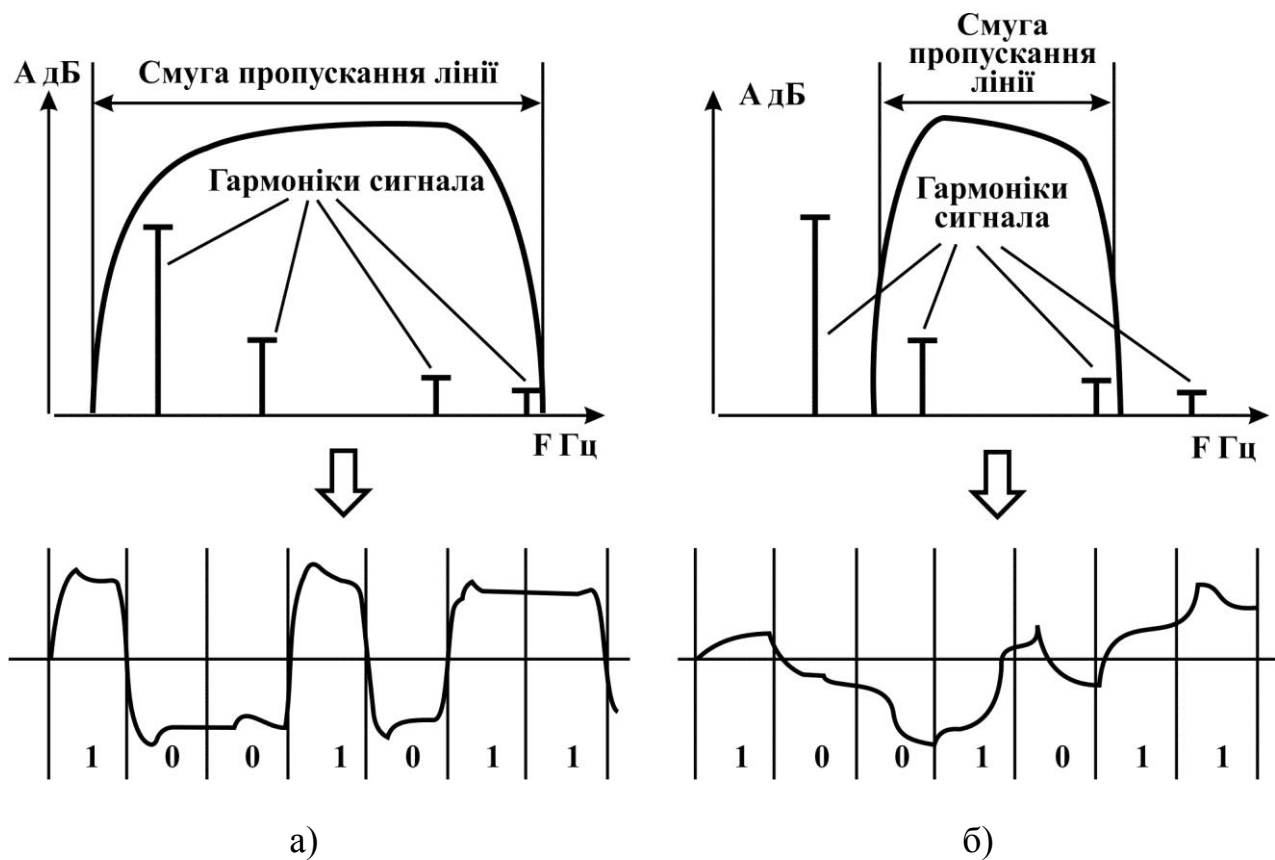


Рисунок 4.9 – Відповідність між смугою пропускання лінії зв'язку й спектром сигналу

Теорія інформації зазначає, що будь-яка помітна й непередбачена зміна прийнятого сигналу несе в собі інформацію. Відповідно до цього прийом синусоїди, у якої амплітуда, фаза й частота залишаються незмінними, інформації не несе, тому що зміна сигналу хоча й відбувається, але є передбачуваною. Аналогічно, не несуть у собі інформації імпульси на тактовій шині комп'ютера, оскільки їхні зміни також постійні в часі. А от імпульси на шині даних передбачити заздалегідь не можна, тому вони переносять інформацію між окремими блоками або пристроями.

Більшість способів кодування використовують зміну будь-якого параметра періодичного сигналу – частоти, амплітуди й фази синусоїди або знак потенціалу послідовності імпульсів. Періодичний сигнал, параметри якого змінюються, називають несучим сигналом, або несучою частотою, якщо як сигнал використовується синусоїда.

Якщо сигнал змінюється так, що можна розрізнити тільки два його стани, то будь-яка його зміна відповідатиме найменшій одиниці інформації – біту. Якщо ж сигнал може мати більше двох помітних станів, то будь-яка його зміна нестиме декілька біт інформації.

Кількість змін інформаційного параметра несучого періодичного сигналу в секунду вимірюється в бодах (baud). Період часу між сусідніми змінами інформаційного сигналу називається тактом роботи передавача.

Пропускна здатність лінії в бітах у секунду в загальному випадку не збігається з кількістю бод. Вона може бути як вище, так і нижче кількості бод, і це співвідношення залежить від способу кодування.

Якщо сигнал має більше двох помітних станів, то пропускна здатність у бітах у секунду буде вище, ніж кількість бод. Наприклад, якщо інформаційними параметрами є фаза та амплітуда синусоїди, причому розрізняються 4 стани фази в 0, 90, 180 і 270 градусів і два значення амплітуди сигналу, то інформаційний сигнал може мати 8 помітних станів. У цьому випадку модем, який працює зі швидкістю 2400 бод (з тактовою частотою 2400 Гц) передає інформацію зі швидкістю 7200 біт/с, оскільки при одній зміні сигналу передається 3 біта інформації.

Під час використання сигналів з двома помітними станами може спостерігатися зворотна картина. Це часто відбувається тому, що для надійного розпізнавання приймачем користувальницької інформації кожен біт у послідовності кодується за допомогою декількох змін інформаційного параметра несучого сигналу. Наприклад, при кодуванні одиничного значення біта імпульсом позитивної полярності, а нульового значення біта – імпульсом

негативної полярності фізичний сигнал двічі змінює свій стан під час передачі кожного біта. При такому кодуванні пропускна здатність лінії удвічі нижче, ніж кількість бод, передана по лінії.

На пропускну здатність лінії впливає не тільки фізичне, але й логічне кодування. Логічне кодування виконується до фізичного кодування та має на увазі заміну біт вихідної інформації новою послідовністю біт, що несе ту саму інформацію, але, крім цього володіє, додатковими властивостями, наприклад, можливістю для приймальної сторони виявляти помилки в прийнятих даних. Супровід кожного байта вихідної інформації одним бітом парності – це приклад часто застосовуваного способу логічного кодування під час передачі даних за допомогою модемів. Іншим прикладом логічного кодування може слугувати шифрація даних, що забезпечує їхню конфіденційність під час передачі через суспільні канали зв'язку. Під час логічного кодування найчастіше вихідна послідовність біт замінюється довшою послідовністю, тому пропускна здатність каналу відносно корисної інформації при цьому зменшується.

Зв'язок між пропускну здатністю лінії та смугою пропускання

Чим вище частота несучого періодичного сигналу, тим більше інформації в одиницю часу передається по лінії й тим вища пропускна здатність лінії при фіксованому способі фізичного кодування. Однак, з іншого боку, зі збільшенням частоти періодичного несучого сигналу збільшується й ширина спектра цього сигналу, тобто різниця між максимальною й мінімальною частотами того набору синусоїд, які в сумі дадуть обрану для фізичного кодування послідовність сигналів. Лінія передає цей спектр синусоїд з тими перекручуваннями, які визначаються її смугою пропускання. Чим більше невідповідність між смугою пропускання лінії й шириною спектра переданих інформаційних сигналів, тим більше сигнали перекручуються й тим імовірніше помилки в розпізнаванні інформації приймаючою стороною, тобто, швидкість передачі інформації насправді виявляється меншою, ніж можна було припустити.

Зв'язок між смугою пропускання лінії і її максимально можливою пропускну здатністю, поза залежністю від прийнятого способу фізичного кодування, встановив Клод Шеннон:

$$C = F \cdot \log_2 \left(1 + \frac{P_C}{P_{\text{ш}}} \right),$$

де C – максимальна пропускна здатність лінії в бітах у секунду,

F – ширина смуги пропускання лінії в герцах,

P_C – потужність сигналу,

$P_{\text{ш}}$ – потужність шуму.

Із цього співвідношення видно, що хоча теоретичної межі пропускну здатності лінії з фіксованою смугою пропускання не існує, на практиці така межа є. Дійсно, підвищити пропускну здатність лінії можна за рахунок збільшення потужності передавача або ж зменшення потужності шуму (завад) на лінії зв'язку. Обидві ці складові піддаються зміні з великими труднощами. Підвищення потужності передавача веде до значного збільшення його габаритів і вартості. Зниження рівня шуму потребує застосування спеціальних кабелів з гарними захисними екранами, що досить дорого, а також зниження шуму в передавачі й проміжній апаратурі, чого досягти досить не просто. До того ж вплив потужностей корисного сигналу й шуму на пропускну здатність обмежено логарифмічною залежністю, яка зростає не так швидко, як прямо-пропорційна. Так, при досить типовому вихідному відношенні потужності сигналу до потужності шуму в 100 разів підвищення потужності передавача удвічі дасть тільки 15 % збільшення пропускну здатності лінії.

Близьким по суті до формули Шеннона є співвідношення, отримане Найквістом, яке також визначає максимально можливу пропускну здатність лінії зв'язку, але без урахування шуму на лінії:

$$C = F \cdot \log_2 M,$$

де M – кількість помітних станів інформаційного параметра.

Якщо сигнал має два помітні стани, то пропускну здатність дорівнює подвоєному значенню ширини смуги пропускання лінії зв'язку (рис. 4.10, а).

Якщо ж передавач використовує більше, ніж 2 стійкі стани сигналу для кодування даних, то пропускну здатність лінії підвищується, оскільки за один такт роботи передавач передає декілька біт вихідних даних, наприклад, 2 біти за наявності чотирьох помітних станів сигналу (рис. 4.10, б).

Хоча формула Найквіста явно не враховує наявності шуму, побічно його вплив відбивається у виборі кількості станів інформаційного сигналу. Для підвищення пропускну здатності каналу хотілося б збільшити цю кількість до значних величин, але на практиці ми не можемо цього зробити через шум на лінії. Отже, для приклада, наведеного на рис. 4.10, можна збільшити пропускну здатність лінії ще удвічі, використавши для кодування даних не 4, а 16 рівнів. Однак якщо амплітуда шуму часто перевищує різницю між сусідніми шістнадцятьма рівнями, то приймач не зможе стійко розпізнавати передані дані. Тому кількість можливих станів сигналу фактично обмежується співвідношенням потужності сигналу й шуму,

а формула Найквіста визначає граничну швидкість передачі даних у тому випадку, коли кількість станів уже обрано з урахуванням можливостей стійкого розпізнавання приймачем.

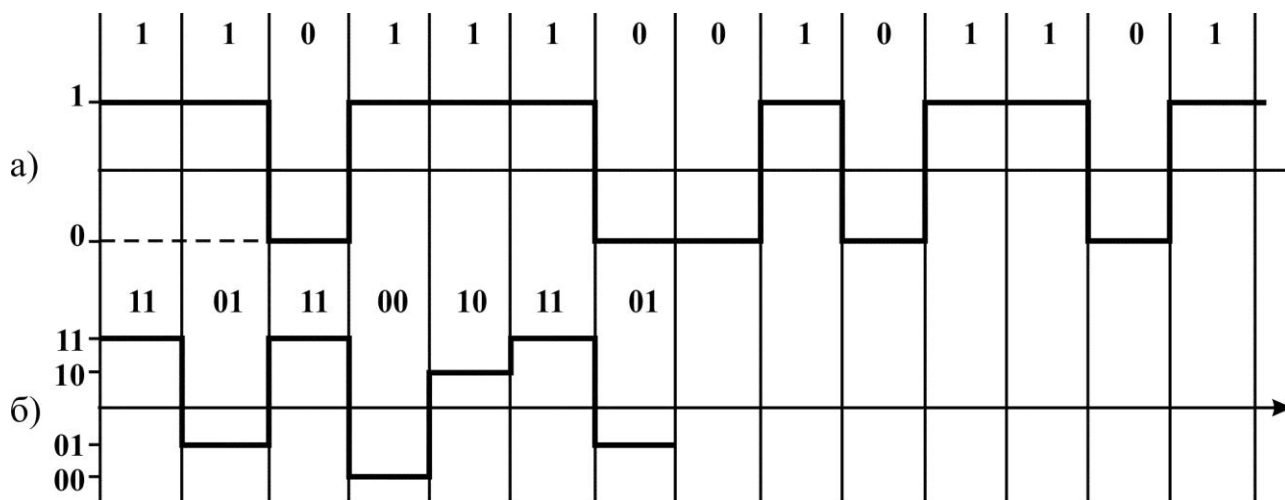


Рисунок 4.10 – Підвищення швидкості передачі за рахунок додаткових станів сигналу

Наведені співвідношення дають граничне значення пропускну здатності лінії, а ступінь наближення до цієї межі залежить від конкретних методів фізичного кодування, розглянутих нижче.

Завадостійкість і вірогідність

Завадостійкість лінії визначає її здатність зменшувати рівень завад, створюваних у зовнішньому середовищі, на внутрішніх провідниках. Завадостійкість лінії залежить від типу використовуваного фізичного середовища, а також від засобів самої лінії, які екранують та пригнічують завади. Найменш завадостійкими є радіолінії, гарну стійкість мають кабельні лінії та відмінну – волоконно-оптичні лінії, малочутливі до зовнішнього електромагнітного випромінювання. Зазвичай для зменшення завад, які з'являються через зовнішні електромагнітні поля, провідники екранують і/або скручують.

Перехресні наведення на ближньому кінці (Near End Cross Talk – NEXT) визначають завадостійкість кабеля до внутрішніх джерел завад, коли електромагнітне поле сигналу, переданого виходом передавача однією парою провідників, наводить на іншу пару провідників сигнал завади. Якщо

до другої пари буде підключений приймач, то він може прийняти наведену внутрішню заваду за корисний сигнал.

Показник NEXT, виражений у децибелах, дорівнює

$$10\log \frac{P_{\text{ВИХ}}}{P_{\text{НАВ}}},$$

де $P_{\text{ВИХ}}$ – потужність вихідного сигналу,

$P_{\text{НАВ}}$ – потужність наведеного сигналу.

Чим менше значення NEXT, тим краще кабель. Так, для крученої пари категорії 5 показник NEXT має бути менше –27 дБ на частоті 100 МГц.

Показник NEXT зазвичай використовується стосовно кабелю, який складається з декількох кручених пар, оскільки в цьому випадку взаємні наведення однієї пари на іншу можуть досягати значних величин. Для одинарного коаксіального кабелю (тобто того, що складається з однієї екранованої жили) цей показник не має сенсу, а для подвійного коаксіального кабелю він також не застосовується внаслідок високого ступеня захищеності кожної жили. Оптичні волокна також не створюють один одному хоч скількись помітних завад.

У зв'язку з тим, що в деяких нових технологіях використовується передача даних водночас декількома крученими парами, останнім часом став застосовуватися показник PowerSUM, який є модифікацією NEXT. Цей показник показує сумарну потужність перехресних наведень від усіх передавальних пар у кабелі.

Вірогідність передачі даних характеризує ймовірність перекручування для кожного переданого біта даних. Іноді цей самий показник називають інтенсивністю бітових помилок (Bit Error Rate, BER). Величина BER для каналів зв'язку без додаткових засобів захисту від помилок (наприклад, кодів, які самокорегуються, або протоколів з повторною передачею перекручених кадрів) становить, як правило, $10^{-4} - 10^{-6}$, в оптоволоконних лініях зв'язку – 10^{-9} . Значення вірогідності передачі даних, наприклад, у 10^{-4} говорить про те, що в середньому з 10000 біт перекручується значення одного біта.

Перекручування біт відбуваються як через наявність завад на лінії, так і через перекручування форми сигналу обмеженою смугою пропускання лінії. Тому для підвищення вірогідності переданих даних слід підвищувати ступінь завадозахищеності лінії, знижувати рівень перехресних наведень у кабелі, а також використовувати більш широкосмугові лінії зв'язку.

4.3 Контрольні запитання та завдання

1. Поясніть, чи можуть цифрові лінії зв'язку передавати аналогові дані?
2. Дайте відповідь, якою буде теоретична межа швидкості передачі даних у бітах у секунду по каналу з шириною смуги пропускання в 20 кГц, якщо потужність передавача становить 0,01 мВт, а потужність шуму в каналі дорівнює 0,0001 мВт?
3. Визначте пропускну здатність каналу зв'язку для кожного з напрямків дуплексного режиму, якщо відомо, що його смуга пропускання дорівнює 600 кГц, а в методі кодування використовується 10 станів сигналу.
4. Що визначають перехресні наведення на ближньому кінці кабелю?
5. Перелічіть методи підвищення швидкості передачі в каналі зв'язку.

Лекція 5

МЕТОДИ ПЕРЕДАЧІ ДИСКРЕТНИХ ДАНИХ НА ФІЗИЧНОМУ РІВНІ

5.1 Стандарти кабелів зв'язку

Кабель – це досить складний виріб, який складається з провідників, шарів екрану та ізоляції. У деяких випадках до складу кабеля входять розніми, за допомогою яких кабелі приєднуються до обладнання. Крім того, для забезпечення швидкої перекомутації кабелів і обладнання використовуються різні електромеханічні пристрої, які називаються кросовими секціями, кросовими коробками або шафами.

У комп'ютерних мережах застосовуються кабелі, які задовольняють певним стандартам, що дозволяє будувати кабельну систему мережі з кабелів і сполучних пристроїв різних виробників. Сьогодні найуживанішими стандартами у світовій практиці є такі:

- американський стандарт EIA/TIA-568A, що був розроблений спільними зусиллями декількох організацій: ANSI, EIA/TIA і лабораторією Underwriters Labs (UL). Стандарт EIA/TIA-568 розроблено на основі попередньої версії стандарту EIA/TIA-568 і доповнень до цього стандарту TSB-36 і TSB-40A);

- міжнародний стандарт ISO/IEC 11801;
- європейський стандарт EN50173.

Ці стандарти близькі між собою й по багатьох позиціях висувають до кабелів ідентичні вимоги. Однак є й розходження між цими стандартами, наприклад, у міжнародний стандарт 11801 і європейський EN50173 увійшли деякі типи кабелів, які відсутні в стандарті EIA/TIA-568A.

До появи стандарту EIA/TIA велике значення мав американський стандарт системи категорій кабелів Underwriters Labs, розроблений разом з компанією Anixter. Пізніше цей стандарт увійшов у стандарт EIA/TIA-568.

Крім цих відкритих стандартів, багато компаній свого часу розробили свої фірмові стандарти, з яких дотепер має практичне значення тільки один – стандарт компанії IBM.

При стандартизації кабелів прийнятий протоково-незалежний підхід. Це означає, що в стандарті визначаються електричні, оптичні та механічні характеристики, яким має задовольняти той чи інший тип кабелю або

сполучного виробу – розніму, кросової коробки тощо. Але для якого протоколу призначений даний кабель, стандарт не визначає. Тому не можна придбати кабель для протоколу Ethernet або FDDI, слід просто знати, які типи стандартних кабелів підтримують протоколи Ethernet і FDDI.

У ранніх версіях стандартів визначалися лише характеристики кабелів, без з'єднувачів. В останніх версіях стандартів з'явилися вимоги до сполучних елементів (документи TSB-36 і TSB-40A, що увійшли потім у стандарт 568A), а також до ліній (каналів), які становлять типове складання елементів кабельної системи, що складається зі шнура від робочої станції до розетки, самої розетки, основного кабелю (довжиною до 90 м для крученої пари), точки переходу (наприклад, ще однієї розетки або жорсткого кросового з'єднання) і шнура до активного обладнання, наприклад концентратора або комутатора.

Ми зупинимося тільки на основних вимогах до самих кабелів, не розглядаючи характеристик сполучних елементів і зібраних ліній.

У стандартах кабелів визначається досить багато характеристик, з яких найважливішими є такі (перші дві уже були детально розглянуті):

- *Загасання (Attenuation)*. Загасання вимірюється в децибелах на метр для певної частоти або діапазону частот сигналів;

- *Перехресні наведення на ближньому кінці (Near End Cross Talk, NEXT)*. Вимірюються в децибелах для певної частоти сигналу;

- *Імпеданс (хвильовий опір)* – це повний (активний і реактивний) опір в електричному колі. Імпеданс вимірюється в Омах і є відносно постійною величиною для кабельних систем (наприклад, для коаксіальних кабелів, використовуваних у стандартах Ethernet, імпеданс кабеля має становити 50 Ом). Для неекранованої крученої пари найчастіше використовувані значення імпедансу – 100 і 120 Ом. В області високих частот (100–200 МГц) імпеданс залежить від частоти;

- *Активний опір* – це опір постійному струму в електричному колі. На відміну від імпедансу активний опір не залежить від частоти й зростає зі збільшенням довжини кабеля.

- *Ємність* – це властивість металевих провідників накопичувати енергію. Два електричних провідники в кабелі, розділені діелектриком, є конденсатором, здатним накопичувати заряд. Ємність – небажана величина, тому слід прагнути до того, щоб вона була якнайменшою (іноді застосовують термін «паразитна ємність»). Високе значення ємності в кабелі призводить до перекручування сигналу й обмежує смугу пропускання лінії.

- *Рівень зовнішнього електромагнітного випромінювання* або електричний шум. Електричний шум – це небажана змінна напруга в провіднику. Електричний шум буває двох типів: фоновий і імпульсний. Електричний шум можна також розділити на низько-, середньо- та високочастотний. Джерелами фонового електричного шуму в діапазоні до 150 кГц є лінії електропередачі, телефони й лампи денного світла; у діапазоні від 150 кГц до 20 МГц – комп'ютери, принтери, ксерокси; у діапазоні від 20 МГц до 1 ГГц – телевізійні й радіопередавачі, мікрохвильові печі. Основними джерелами імпульсного електричного шуму є мотори, перемикачі та зварювальні агрегати. Електричний шум вимірюється в мілівольтах.

- *Діаметр або площа перетину провідника*. Для мідних провідників досить застосовуваною є американська система AWG (American Wire Gauge), що вводить деякі умовні типи провідників, наприклад 22 AWG, 24 AWG, 26 AWG. Чим більше номер типу провідника, тим менше його діаметр. В обчислювальних мережах найуживанішими є типи провідників, наведені вище як приклади. У європейських і міжнародних стандартах діаметр провідника вказується в міліметрах. Природно, що наведений перелік характеристик далеко не повний, причому в ньому подано тільки електромагнітні характеристики і його слід доповнити механічними й конструктивними, які визначають тип ізоляції, конструкцію рознімів тощо. Крім універсальних характеристик, таких, як загасання, що застосовуються для всіх типів кабелів, існують характеристики, які застосовуються лише до певного типу кабелю. Наприклад, параметр крок скрутки проводів використовується лише для характеристики крученої пари, а параметр NEXT – до багатопарних кабелів на основі крученої пари.

Основна увага в сучасних стандартах приділяється кабелям на основі крученої пари й волоконно-оптичним кабелям.

Кабелі на основі неекранованої крученої пари

Мідний неекранований кабель UTP залежно від електричних і механічних характеристик поділяється на 5 категорій (Category I – Category 5). Кабелі категорій 1 і 2 були визначені в стандарті EIA/TIA-568, але в стандарт 568A вже не увійшли, як застарілі.

Кабелі категорії 1 застосовуються там, де вимоги до швидкості передачі мінімальні. Зазвичай це кабель для цифрової та аналогової передачі голосу й низькошвидкісної (до 20 Кбіт/с) передачі даних. До 1983 року це був основний тип кабелю для телефонного розведення.

Кабелі категорії 2 були вперше застосовані фірмою IBM під час побудови власної кабельної системи. Головна вимога до кабелів цієї категорії – здатність передавати сигнали зі спектром до 1 МГц.

Кабелі категорії 3 стандартизовані в 1991 році, коли був розроблений Стандарт телекомунікаційних кабельних систем для комерційних будинків (EIA-568), на основі якого потім було створено чинний стандарт EIA-568A. Стандарт EIA-568 визначив електричні характеристики кабелів категорії 3 для частот у діапазоні до 16 МГц, які підтримують, таким чином, високошвидкісні мережні додатки. Кабель категорії 3 призначений як для передачі даних, так і для передачі голосу. Крок скрутки проводів дорівнює приблизно 3 витки на 1 фут (30,5 см). Кабелі категорії 3 сьогодні становлять основу багатьох кабельних систем будинків, використовуються для передачі як голосу, так і даних.

Кабелі категорії 4 є дещо покращеним варіантом кабелів категорії 3. Кабелі категорії 4 зобов'язані витримувати тести на частоті передачі сигналу 20 МГц і забезпечувати підвищену завадостійкість і низькі втрати сигналу. Кабелі категорії 4 добре підходять для застосування в системах зі збільшеними відстанями (до 135 метрів) і в мережах Token Ring із пропускнуою здатністю 16 Мбіт/с. На практиці використовуються рідко.

Кабелі категорії 5 були спеціально розроблені для підтримки високошвидкісних протоколів. Тому їхні характеристики визначаються в діапазоні до 100 МГц. Більшість нових високошвидкісних стандартів орієнтуються на використання крученої пари 5 категорії. На цьому кабелі працюють протоколи зі швидкістю передачі даних 100 Мбіт/с – FDDI (з фізичним стандартом TP-PMD), Fast Ethernet, 100VG-AnyLAN, а також більш швидкісні протоколи – ATM на швидкості 155 Мбіт/с, і Gigabit Ethernet на швидкості 1000 Мбіт/с (варіант Gigabit Ethernet на крученій парі категорії 5 став стандартом у червні 1999 р.). Кабель категорії 5 прийшов на заміну кабелю категорії 3, і сьогодні всі нові кабельні системи великих будинків базуються саме на цьому типі кабелю (у поєднанні з волоконно-оптичним).

Найбільш важливі електромагнітні характеристики кабелю категорії 5 мають такі значення:

- повний хвильовий опір у діапазоні частот до 100 МГц дорівнює 100 Ом (стандарт ISO 11801 допускає також кабель з хвильовим опором 120 Ом);
- величина перехресних наведень NEXT залежно від частоти сигналу має приймати значення не менші 74 дБ на частоті 150 кГц і не менші 32 дБ на частоті 100 МГц;

- загасання має граничні значення від 0,8 дБ (на частоті 64 кГц) до 22 дБ (на частоті 100 МГц);
- активний опір не має перевищувати 9,4 Ом на 100 м;
- ємність кабелю не має перевищувати 5,6 нФ на 100 м.

Усі кабелі UTP незалежно від їхньої категорії випускаються в чотирипарному виконанні. Кожна із чотирьох пар кабеля має певний колір і крок скрутки. Зазвичай дві пари призначені для передачі даних, а дві – для передачі голосу.

Для з'єднання кабелів з обладнанням використовуються вилки й розетки RJ-45, які є восьмиконтактними рознімами, схожими на звичайні телефонні розніми, RJ-11.

Особливе місце займають кабелі категорій 6 і 7, які промисловість порівняно недавно почала випускати. Для кабелю категорії 6 характеристики визначаються до частоти 200 МГц, а для кабелів категорії 7 – до 600 МГц. Кабелі категорії 7 обов'язково екрануються, причому як кожна пара, так і весь кабель у цілому. Кабель категорії 6 може бути як екранованим, так і неекранованим. Основне призначення цих кабелів – підтримка високошвидкісних протоколів на відрізках кабелю більшої довжини, ніж кабель UTP категорії 5. Деякі фахівці сумніваються в необхідності застосування кабелів категорії 7, оскільки вартість кабельної системи під час їхнього використання виходить порівняною за вартістю мережі з використанням волоконно-оптичних кабелів, а характеристики кабелів на основі оптичних волокон вище.

Кабелі на основі екранованої крученої пари

Екранована кручена пара STP добре захищає передані сигнали від зовнішніх завад, а також менше випромінює електромагнітних коливань зовні, що захищає, у свою чергу, користувачів мереж від шкідливого для здоров'я випромінювання. Наявність екрана, що заземлюється, здорожує кабель і ускладнює його прокладання, оскільки вимагає виконання якісного заземлення. Екранований кабель застосовується тільки для передачі даних, а голос по ньому не передають.

Основним стандартом, який визначає параметри екранованої крученої пари, є фірмовий стандарт IBM. У цьому стандарті кабелі поділяються не на категорії, а на типи: Type 1, Type 2,..., Type 9.

Основним типом екранованого кабелю є кабель Type 1 стандарту IBM. Він складається з двох пар скручених проводів, екранованих провідною

обплетенням, що заземлюється. Електричні параметри кабелю Type 1 приблизно відповідають параметрам кабелю UTP категорії 5. Однак хвильовий опір кабелю Type 1 дорівнює 150 Ом (UTP категорії 5 має хвильовий опір 100 Ом), тому просте «покращення» кабельної проводки мережі шляхом заміни неекранованої пари UTP на STP Type 1 неможливо. Трансивери, розраховані на роботу з кабелем, який має хвильовий опір 100 Ом, погано працюватимуть на хвильовий опір 150 Ом. Тому, під час використання STP Type 1 необхідні відповідні трансивери. Такі трансивери є в мережних адаптерах Token Ring, оскільки ці мережі розроблялися для роботи на екранованій кручений парі. Деякі інші стандарти також підтримують кабель STP Type 1 – наприклад, 100VG-AnyLAN, а також Fast Ethernet (хоча основним типом кабелю для Fast Ethernet є UTP категорії 5). У випадку, якщо технологія може використовувати UTP і STP, слід переконатися, на який тип кабелю розраховані придбані трансивери. Сьогодні кабель STP Type 1 включений у стандарти EIA/TIA-568A, ISO 11801 і EN50173, тобто набув міжнародного статусу.

Екрановані кручені пари використовуються також у кабелі IBM Type 2, що становить кабель Type 1 з доданими двома парами неекранованого проводу для передачі голосу.

Для приєднання екранованих кабелів до обладнання використовуються різні конструкції IBM.

Не всі типи кабелів стандарту IBM належать до екранованих кабелів – деякі визначають характеристики неекранованого телефонного кабелю (Type 3) і оптоволоконного кабелю (Type 5).

Коаксіальні кабелі

Існує велика кількість типів коаксіальних кабелів, які використовуються у мережах різного типу – телефонних, телевізійних і комп'ютерних. Нижче наводяться основні типи й характеристики цих кабелів.

- RG-8 і RG-11 – «товстий» коаксіальний кабель, розроблений для мереж Ethernet 10Base-5. Має хвильовий опір 50 Ом і зовнішній діаметр 0,5 дюйма (близько 12 мм). Цей кабель має досить товстий внутрішній провідник діаметром 2,17 мм, який забезпечує гарні механічні та електричні характеристики (загасання на частоті 10 МГц – не гірше 18 дБ/км). Але цей кабель складно монтувати – він погано гнеться.

- RG-58/U, RG-58 A/U і RG-58 C/U – різновид «тонкого» коаксіального кабелю для мереж Ethernet 10Base-2. Кабель RG-58/U має суцільний внутрішній провідник, а кабель RG-58 A/U – багатожильний. Кабель RG-58 C/U проходить

«військове приймання». Всі ці різновиди кабелю мають хвильовий опір 50 Ом, але гірші механічні та електричні характеристики порівняно з «товстим» коаксіальним кабелем. Тонкий внутрішній провідник 0,89 мм не такий міцний, але має набагато більшу гнучкість, зручну під час монтажу. Загасання в цьому типі кабелю вище, ніж в «товстому» коаксіальному кабелі, що призводить до необхідності зменшувати довжину кабелю для отримання однакового загасання в сегменті. Для з'єднання кабелів з обладнанням використовується рознім типу BNC.

- RG-59 – телевізійний кабель з хвильовим опором 75 Ом. Широко застосовується в кабельному телебаченні.

- RG-62 – кабель з хвильовим опором 93 Ом, використовується в мережах ArcNet, обладнання яких сьогодні практично не випускається.

Коаксіальні кабелі із хвильовим опором 50 Ом (тобто «тонкий» і «товстий») описані в стандарті EIA/TIA-568. Новий стандарт EIA/TIA-568A коаксіальні кабелі не описує, як морально застарілі.

Волоконно-оптичні кабелі

Волоконно-оптичні кабелі складаються з центрального провідника світла (серцевини) – скляного волокна, оточеного другим шаром скла – оболонкою, яка має менший показник переломлення, ніж серцевина. Поширюючись у серцевині, промені світла не виходять за її межі, відбиваючись від покриваючого шару оболонки. Залежно від розподілу показника переломлення й від величини діаметра сердечника розрізняють:

- багатомодове волокно зі східчастою зміною показника заломлення (рис. 5.1, а);

- багатомодове волокно із плавною зміною показника заломлення (рис. 5.1, б);

- одномодове волокно (рис. 5.1, в).

Поняття «мода» описує режим поширення світлових променів у внутрішньому сердечнику кабелю. В одномодовому кабелі (Single Mode Fiber, SMF) використовується центральний провідник дуже малого діаметра, порівняного з довжиною хвилі світла – від 5 до 10 мкм. При цьому практично всі промені світла поширюються уздовж оптичної осі світловоду, не відбиваючись від зовнішнього провідника. Смуга пропускання одномодового кабелю дуже широка – до сотень гігагерц на кілометр. Виготовлення тонких якісних волокон для одномодового кабелю є складним технологічним процесом, що робить одномодовий кабель досить дорогим. Крім того,

у волокно такого маленького діаметра досить складно направити пучок світла, не втративши при цьому значну частину його енергії.

У багатомодових кабелях (Multi Mode Fiber, MMF) використовуються ширші внутрішні сердечники, які легше виготовити технологічно. У стандартах визначені два найбільш уживаних багатомодових кабелі: 62,5/125 мкм і 50/125 мкм, де 62,5 мкм або 50 мкм – це діаметр центрального провідника, а 125 мкм – діаметр зовнішнього провідника.

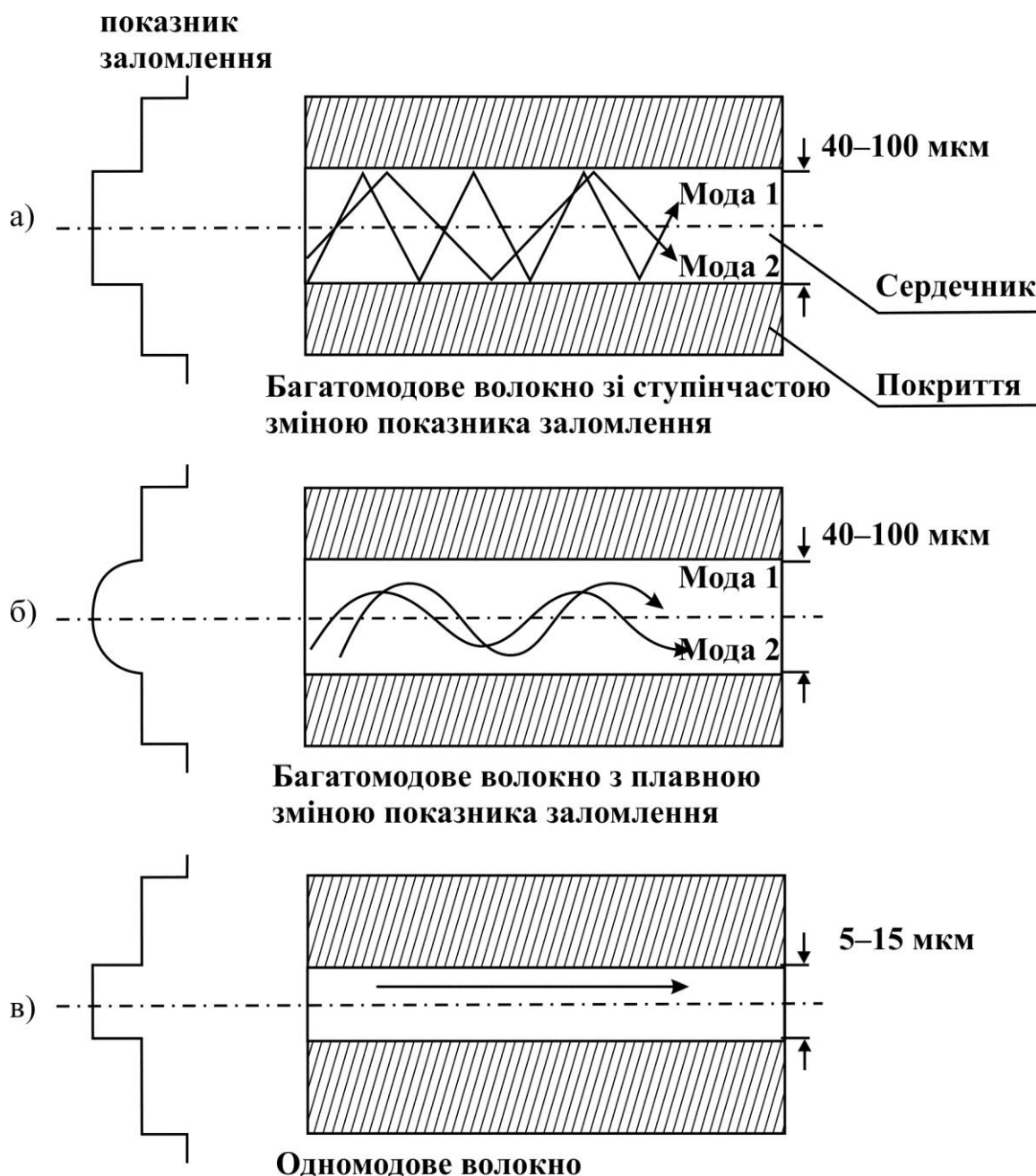


Рисунок 5.1 – Типи оптичного кабелю

У багатомодових кабелях у внутрішньому провіднику водночас існує кілька світлових променів, які відбиваються від зовнішнього провідника під різними кутами. Кут відбиття променя називається модою променя.

У багатомодових кабелях із плавною зміною коефіцієнта переломлення режим поширення кожної моди має більш складний характер.

Багатомодові кабелі мають вузьку смугу пропускання – від 500 до 800 МГц/км. Звуження смуги відбувається через втрати світлової енергії під час відбиття, а також через інтерференцію променів різних мод.

Як джерела випромінювання світла у волоконно-оптичних кабелях застосовуються:

- світлодіоди;
- напівпровідникові лазери.

Для одномодових кабелів застосовуються тільки напівпровідникові лазери, оскільки при такому малому діаметрі оптичного волокна світловий потік, що створюється світлодіодом, не можливо без великих втрат направити у волокно. Для багатомодових кабелів використовуються дешевші світлодіодні випромінювачі.

Для передачі інформації застосовується світло з довжиною хвилі 1550 нм (1,55 мкм), 1300 нм (1,3 мкм) і 850 нм (0,85 мкм). Світлодіоди можуть випромінювати світло з довжиною хвилі 850 нм і 1300 нм. Випромінювачі з довжиною хвилі 850 нм істотно дешевші, ніж випромінювачі з довжиною хвилі 1300 нм, але смуга пропускання кабеля для хвиль 850 нм вузька, наприклад, 200 МГц/км замість 500 МГц/км.

Лазерні випромінювачі працюють на довжинах хвиль 1300 і 1550 нм. Швидкодія сучасних лазерів дозволяє модулювати світловий потік із частотами 10 ГГц і вище. Лазерні випромінювачі створюють когерентний потік світла, за рахунок чого втрати в оптичних волокнах стають менше, ніж під час використання некогерентного потоку світлодіодів.

Використання тільки декількох довжин хвиль для передачі інформації в оптичних волокнах пов'язано з особливістю їхньої амплітудно-частотної характеристики. Саме для цих дискретних довжин хвиль спостерігаються яскраво виражені максимуми передачі потужності сигналу, а для інших хвиль загасання у волокнах істотно вище.

Волоконно-оптичні кабелі приєднують до обладнання рознімами MLC, ST і SC.

Волоконно-оптичні кабелі мають чудові характеристики всіх типів: електромагнітні, механічні (добре гнуться, а у відповідній ізоляції мають гарну механічну міцність). Однак у них є один серйозний недолік – складність з'єднання волокон з рознімами й між собою за необхідності нарощування довжини кабелю.

Сама вартість волоконо-оптичних кабелів ненабагато перевищує вартість кабелів на крученій парі, однак проведення монтажних робіт з оптоволоконном обходиться набагато дорожче через трудомісткість операцій і високу вартість монтажного обладнання, яке застосовується. Так, приєднання оптичного волокна до розніму потребує проведення високоточної обрізки волокна в площині строго перпендикулярній осі волокна, а також виконання з'єднання шляхом складної операції склеювання, а не обтиснення, як це робиться для крученої пари. Виконання ж неякісних з'єднань одразу різко звужує смугу пропускання волоконно-оптичних кабелів і ліній.

5.2 Методи аналогової модуляції

Під час передачі дискретних даних по каналах зв'язку застосовуються два основних типи фізичного кодування – на основі синусоїдального несучого сигналу й на основі послідовності прямокутних імпульсів. Перший спосіб часто називається також модуляцією або аналоговою модуляцією, підкреслюючи той факт, що кодування здійснюється за рахунок зміни параметрів аналогового сигналу. Другий спосіб зазвичай називають цифровим кодуванням. Ці способи відрізняються шириною спектра результуючого сигналу й складністю апаратури, необхідної для їхньої реалізації.

В ході використання прямокутних імпульсів спектр результуючого сигналу виходить досить широким. Це не дивно, якщо згадати, що спектр ідеального імпульсу має нескінченну ширину. Застосування синусоїди приводить до спектра набагато меншої ширини при тій же швидкості передачі інформації. Однак для реалізації синусоїдальної модуляції потрібна більш складна й дорога апаратура, ніж для реалізації прямокутних імпульсів.

На сьогодні все частіше дані, які спочатку мають аналогову форму – мова, телевізійне зображення, – передаються по каналах зв'язку в дискретному вигляді, тобто у вигляді послідовності одиниць і нулів. Процес подання аналогової інформації в дискретній формі називається дискретною модуляцією. Терміни «модуляція» і «кодування» часто використовують як синоніми.

Аналогова модуляція

Аналогова модуляція застосовується для передачі дискретних даних по каналах з вузькою смугою частот, типовим представником яких є канал тональної частоти, що надається у розпорядження користувачам громадських телефонних мереж. Типова амплітудно-частотна характеристика каналу

тональної частоти подана на рис. 5.2. Цей канал передає частоти в діапазоні від 300 до 3400 Гц, отже, його смуга пропускання дорівнює 3100 Гц. Хоча людський голос має набагато ширший спектр – приблизно від 100 Гц до 10 кГц, – для прийнятної якості передачі мови діапазон в 3100 Гц є гарним рішенням. Чітке обмеження смуги пропускання тонального каналу пов'язано з використанням апаратури ущільнення й комутації каналів у телефонних мережах.

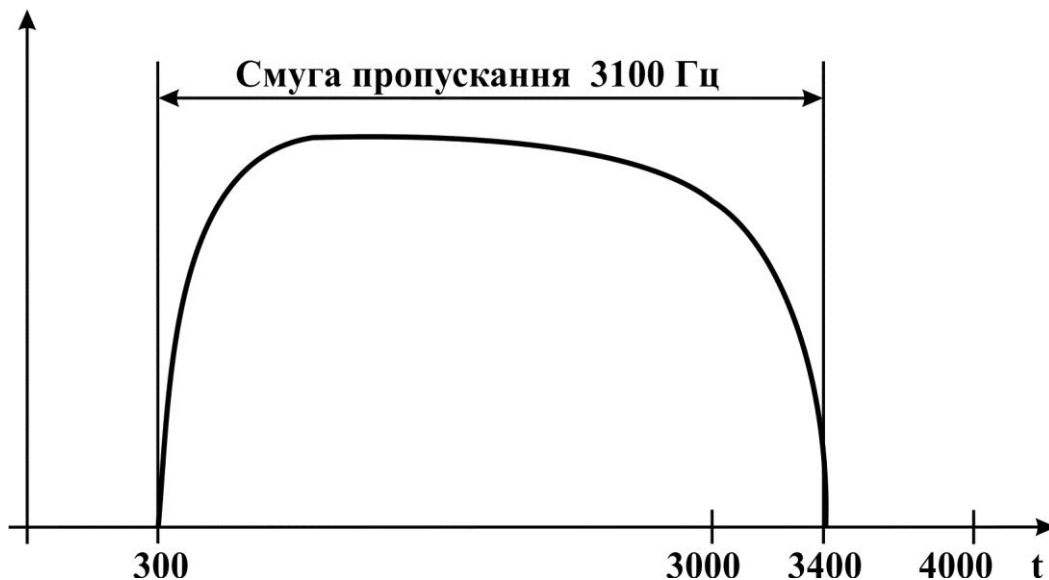


Рисунок 5.2 – Амплітудно-частотна характеристика каналу тональної частоти

Пристрій, що виконує функції модуляції несучої синусоїди на передавальній стороні й демодуляції на приймальній стороні, називається модемом (модулятор/демодулятор).

Методи аналогової модуляції

Аналогова модуляція є таким способом фізичного кодування, під час якого інформація кодується зміною амплітуди, частоти або фази синусоїдального сигналу несучої частоти. Основні способи аналогової модуляції показані на рис. 5.3. На діаграмі (рис. 5.3, а) показана послідовність біт вихідної інформації, подана потенціалами високого рівня для логічної одиниці й потенціалом нульового рівня для логічного нуля. Такий спосіб кодування називається потенційним кодом, який часто використовується в ході передачі даних між блоками комп'ютера.

При амплітудній модуляції (рис. 5.3, б) для логічної одиниці обирається один рівень амплітуди синусоїди несучої частоти, а для логічного нуля – іншої. Цей спосіб на практиці рідко використовується в чистому вигляді через низьку

завадостійкість, але часто застосовується в поєднанні з іншим виглядом модуляції – фазовою модуляцією.

Під час частотної модуляції (рис. 5.3, в) значення 0 і 1 вихідних даних передаються синусоїдами з різною частотою – f_0 і f_1 . Цей спосіб модуляції не вимагає складних схем у модемах і зазвичай застосовується в низькошвидкісних модемах, які працюють на швидкостях 300 або 1200 біт/с.

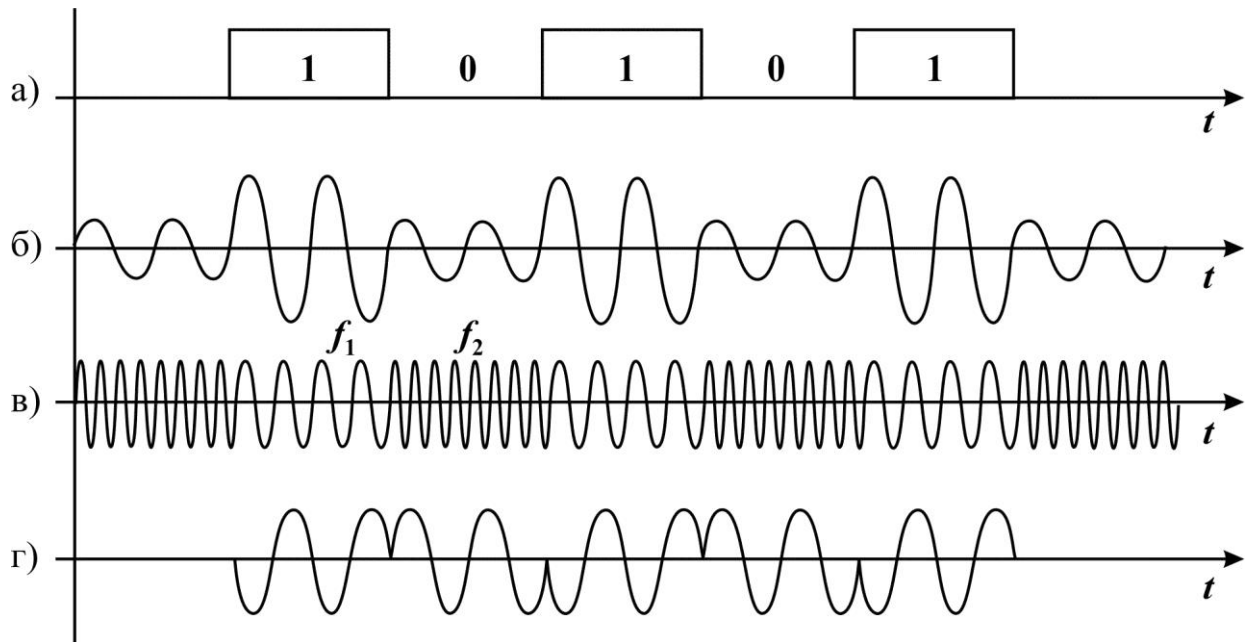


Рисунок 5.3 – Різні типи модуляції

Під час фазової модуляції (рис. 5.3, г) значенням даних 0 і 1 відповідають сигнали однакової частоти, але з різною фазою, наприклад 0 і 180 градусів або 0, 90, 180 і 270 градусів.

У швидкісних модемах часто використовуються комбіновані методи модуляції, як правило, амплітудна в поєднанні з фазовою.

Спектр модульованого сигналу

Спектр результуючого модульованого сигналу залежить від типу й швидкості модуляції, тобто бажаної швидкості передачі біт вихідної інформації.

Розглянемо спочатку спектр сигналу при потенційному кодуванні. Нехай логічна одиниця кодується позитивним потенціалом, а логічний нуль – негативним потенціалом такої ж величини. Для спрощення обчислень припустимо, що передається інформація, яка складається з нескінченної послідовності одиниць і нулів, як це показано на рис. 5.3, а. Зазначимо, що в цьому випадку величини бод і біт у секунду збігаються.

Для потенційного кодування спектр виходить із формул Фур'є для періодичної функції. Якщо дискретні дані передаються з бітовою швидкістю N біт/с, то спектр складається з постійної складової нульової частоти та нескінченного ряду гармонік з частотами $f_0, 3f_0, 5f_0, 7f_0, \dots$, де $f_0 = N/2$. Амплітуди цих гармонік убивають досить повільно – з коефіцієнтами $1/3, 1/5, 1/7, \dots$ від амплітуди гармоніки f_0 (рис. 5.4, а). У результаті спектр потенційного коду потребує для якісної передачі широку смугу пропускання. Крім того, слід урахувати, що реально спектр сигналу постійно змінюється залежно від того, які дані передаються по лінії зв'язку. Наприклад, передача довгої послідовності нулів або одиниць зрушує спектр у бік низьких частот, а в крайньому випадку, коли передані дані складаються тільки з одиниць (або тільки з нулів), спектр складається з гармоніки нульової частоти. Під час передачі одиниць і нулів, які чергуються, постійна складова відсутня. Тому спектр результуючого сигналу потенційного коду під час передачі довільних даних займає смугу від деякої величини, близької до 0 Гц, до приблизно $7f_0$ (гармоніками із частотами вище $7f_0$ можна зневажити через малий внесок у результуючий сигнал). Для каналу тональної частоти верхня межа під час потенційного кодування досягається для швидкості передачі даних в 971 біт/с, а нижня неприйнятна для будь-яких швидкостей, оскільки смуга пропускання каналу починається з 300 Гц. У результаті потенційні коди на каналах тональної частоти ніколи не використовуються.

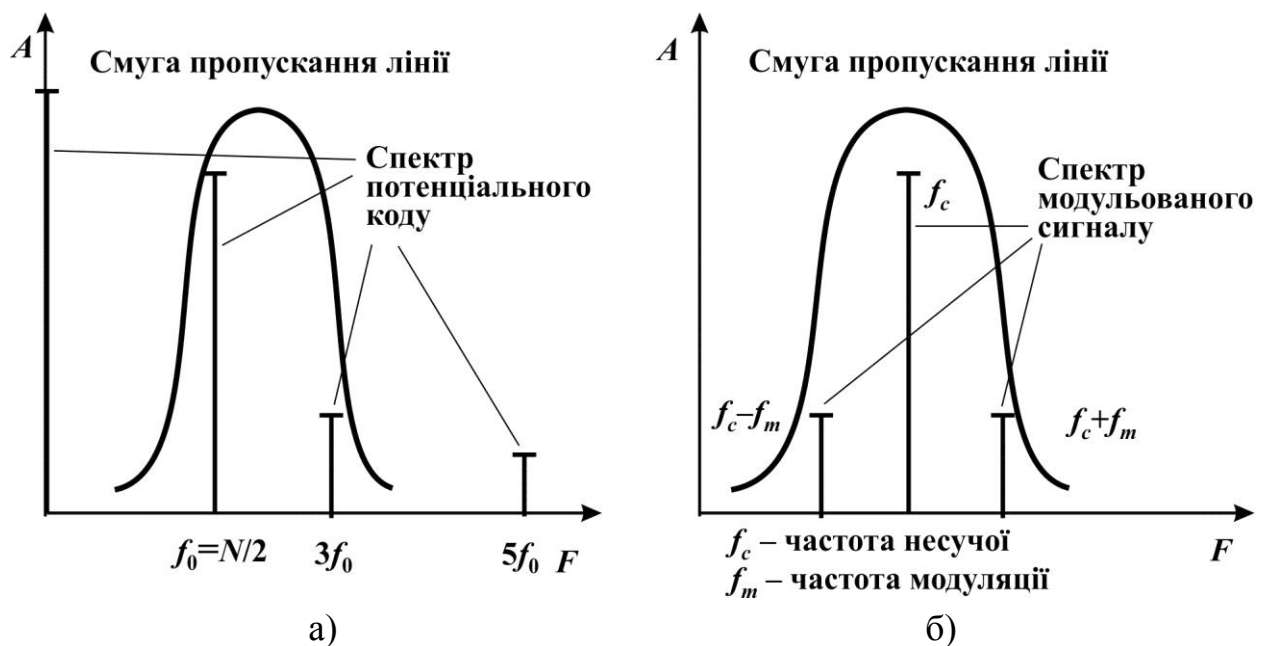


Рисунок 5.4 – Спектри сигналів під час потенційного кодування й амплітудної модуляції

Під час амплітудної модуляції спектр складається із синусоїди несучої частоти f_c і двох бічних гармонік: $(f_c + f_m)$ і $(f_c - f_m)$, де f_m – частота зміни інформаційного параметра синусоїди, що збігається зі швидкістю передачі даних під час використання двох рівнів амплітуди (рис. 5.4, б). Частота f_m визначає пропускну здатність лінії для даного способу кодування. При невеликій частоті модуляції ширина спектра сигналу буде також невеликою (дорівнюватиме $2f_m$), тому сигнали не перекручуватимуться лінією, якщо її смуга пропускання буде більшою або дорівнюватиме $2f_m$. Для каналу тональної частоти такий спосіб модуляції прийнятний при швидкості передачі даних не більше $3100/2 = 1550$ біт/с. Якщо ж для подання даних використовуються чотири рівні амплітуди, то пропускну здатність каналу підвищується до 3100 біт/с.

При фазовій і частотній модуляції спектр сигналу виходить складнішим, ніж при амплітудній модуляції, оскільки бічних гармонік тут утворюється більше двох, але вони також симетрично розташовані щодо основної несучої частоти, а їхні амплітуди швидко спадають. Тому ці види модуляції також добре підходять для передачі даних по каналу тональної частоти.

Для підвищення швидкості передачі даних використовують комбіновані методи модуляції. Найпоширенішими є методи квадратурної амплітудної модуляції (Quadrature Amplitude Modulation, QAM). Ці методи засновані на комбінації фазової модуляції з 8 значеннями величин зсуву фази й амплітудної модуляції з 4 рівнями амплітуди. Однак з можливих 32 комбінацій сигналу використовуються далеко не всі. Наприклад, у кодах Трелліса припустимі всього 6, 7 або 8 комбінацій для подання вихідних даних, а інші комбінації заборонені. Така надмірність кодування потрібна для розпізнавання модемом помилкових сигналів, які є наслідком перекручувань через завади, які на телефонних каналах, особливо комутованих, досить значні за амплітудою і тривалі у часі.

Висновки

1. Під час побудови мереж застосовуються лінії зв'язку, які використовують різне фізичне середовище: телефонний та телеграфний провід, підвішені в повітрі, мідні коаксіальні кабелі, мідні кручені пари, волоконно-оптичні кабелі, радіохвилі.

2. Лінії зв'язку можуть використовувати, крім кабелю, проміжну апаратуру, прозору для користувачів. Проміжна апаратура виконує дві основні

функції: підсилює сигнали й забезпечує постійну комутацію між парою користувачів лінії.

3. Залежно від типу проміжної апаратури лінії зв'язку поділяються на аналогові й цифрові. В аналогових лініях зв'язку для ущільнення низькошвидкісних каналів абонентів у загальний високошвидкісний канал використовується метод поділу частот (FDM), а в цифрових – метод поділу в часі (TDM).

4. Для характеристики здатності лінії передавати сигнали довільної форми без значних перекручувань застосовується ряд показників, які використовують як тестовий сигнал синусоїди різної частоти. До цих показників належать: амплітудно-частотна характеристика, смуга пропускання й загасання сигналу на визначеній частоті.

5. У комп'ютерних мережах застосовуються кабелі, які відповідають певним стандартам. Сучасні стандарти визначають характеристики не окремого кабелю, а повного набору елементів, необхідного для створення кабельного з'єднання, наприклад шнура від робочої станції до розетки, самої розетки, основного кабелю, жорсткого кросового з'єднання й шнура до концентратора. Сьогодні найуживанішими стандартами є: американський стандарт EIA/TIA-568A, міжнародний стандарт ISO/IEC 11801, європейський стандарт EN50173, а також фірмовий стандарт компанії IBM.

6. Стандарти визначені для чотирьох типів кабелю: на основі неекранованої крученої пари, на основі екранованої крученої пари, коаксіального й волоконно-оптичного кабелів.

7. Кабель на основі неекранованої крученої пари залежно від електричних і механічних характеристик поділяється на 5 категорій. Кабелі категорії 1 застосовуються там, де вимоги до швидкості передачі мінімальні. Головна особливість кабелів категорії 2 – здатність передавати сигнали зі спектром до 1 МГц. Кабелі категорії 3 широко поширені й призначені як для передачі даних, так і для передачі голосу. Кабелі категорії 4 є дещо покращеним варіантом кабелів категорії 3 і на практиці використовуються рідко. Кабелі категорії 5 були спеціально розроблені для підтримки високошвидкісних протоколів FDDI, Fast Ethernet, 100VG-AnyLAN, ATM і Gigabit Ethernet.

8. Кабель на основі екранованої крученої пари добре захищає передані сигнали від зовнішніх завад, а користувачів мереж – від шкідливого для здоров'я випромінювання. Наявність екрана, який заземлюється, здорожує кабель і ускладнює його прокладання. Екранований кабель застосовується тільки для передачі даних. Основним стандартом, який визначає параметри

екранованої крученої пари, є фірмовий стандарт IBM, У цьому стандарті кабелі діляться на типи: Type 1, Type 2, ... Type 9, з яких основним є кабель Type 1.

9. Коаксіальних кабелів є велика кількість варіантів: «товстий» коаксіальний кабель, багато різновидів «тонкого» коаксіального кабелю, які мають гірші механічні й електричні характеристики порівняно з «товстим» коаксіальним кабелем, але за рахунок своєї гнучкості зручніші в ході монтажу, сюди ж належить телевізійний кабель.

10. Волоконно-оптичні кабелі мають чудові електромагнітні й механічні характеристики, але їхнім недоліком є складність і висока вартість монтажних робіт.

5.3 Контрольні запитання та завдання

1. Розрахуйте затримку поширення сигналу й затримку передачі даних для випадку передачі пакета в 128 байт:

- кабелем крученої пари довжиною в 100 м при швидкості передачі 100 Мбіт/с;

- коаксіальним кабелем довжиною в 2 км при швидкості передачі в 10 Мбіт/с;

- супутниковим геостационарним каналом довжиною в 72000 км при швидкості передачі 128 Кбіт/с.

Припустити, що швидкість поширення сигналу дорівнює швидкості світла у вакуумі 300000 км/с.

2. Поясніть, чим обмежується застосування волоконно-оптичних кабелів для побудови мереж ліній зв'язку?

3. Назвіть, що обумовлюють сучасні стандарти кабелів зв'язку?

4. Назвіть, що входить до складу апаратури передачі даних?

Лекція 6

МЕТОДИ ЦИФРОВОГО КОДУВАННЯ

6.1 Вимоги до методів цифрового кодування

Для цифрового кодування дискретної інформації застосовують потенційні й імпульсні коди.

У потенційних кодах для подання логічних одиниць і нулів використовується тільки значення потенціалу сигналу, а його перепади, які формують закінчені імпульси, до уваги не беруться. Імпульсні коди дозволяють подати двійкові дані або імпульсами певної полярності, або частиною імпульсу – перепадом потенціалу певного напрямку.

Під час використання прямокутних імпульсів для передачі дискретної інформації необхідно обрати такий спосіб кодування, який досягав би декількох цілей водночас:

- мав при одній і тій самій бітовій швидкості найменшу ширину спектра результуючого сигналу;
- забезпечував синхронізацію між передавачем і приймачем;
- мав здатність розпізнавати помилки;
- мав низьку вартість реалізації.

Більш вузький спектр сигналів дозволяє на одній і тій же лінії (з однією й тією самою смугою пропускання) домагатися більш високої швидкості передачі даних. Крім того, часто до спектра сигналу висувається вимога відсутності постійної складової, тобто наявності постійного струму між передавачем і приймачем, зокрема, застосування різних трансформаторних схем гальванічної розв'язки перешкоджає проходженню постійного струму.

Синхронізація передавача й приймача потрібна для того, щоб приймач точно знав, у який момент часу необхідно зчитувати нову інформацію з лінії зв'язку. Ця проблема в мережах вирішується складніше, ніж під час обміну даними між близько розташованими пристроями, наприклад, між блоками всередині комп'ютера або між комп'ютером і принтером. На невеликих відстанях добре працює схема, заснована на окремій тактуючій лінії зв'язку (рис. 6.1), так що інформація знімається з лінії даних тільки в момент приходу тактового імпульсу. У мережах використання цієї схеми викликає труднощі через неоднорідність характеристик провідників у кабелях. На великих відстанях нерівномірність швидкості поширення сигналу може

призвести до того, що тактовий імпульс прийде настільки пізніше або раніше відповідного сигналу даних, що біт даних буде пропущений або зчитаний повторно. Другою причиною, через яку в мережах відмовляються від використання тактуючих імпульсів, є економія провідників у дорогих кабелях.

Тому в мережах застосовуються так звані коди, що самосинхронізуються, сигнали яких несуть для передавача позначення, у який момент часу потрібно здійснювати розпізнавання чергового біта (або декількох біт, якщо код орієнтований більш ніж на два стани сигналу). Будь-який різкий перепад сигналу – так званий «фронт» – може слугувати гарним позначенням для синхронізації приймача з передавачем.

Під час використання синусоїд як несучого сигналу результуючий код має властивість самосинхронізації, оскільки зміна амплітуди несучої частоти дає можливість приймачу визначити момент появи вхідного коду.

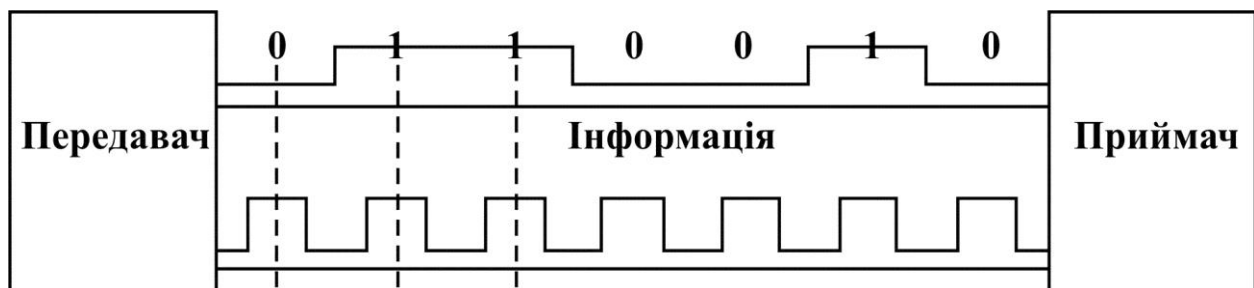


Рисунок 6.1 – Синхронізація приймача й передавача на невеликих відстанях

Розпізнавання й корекцію перекручених даних складно здійснити засобами фізичного рівня, тому найчастіше цю роботу беруть на себе протоколи, які лежать вище: канальний, мережний, транспортний або прикладний. З іншого боку, розпізнавання помилок на фізичному рівні заощаджує час, оскільки приймач не очікує повного поміщення кадру в буфер, а відбраковує його одразу під час розпізнавання помилкових біт усередині кадру.

Вимоги, пропоновані до методів кодування, взаємно суперечливі, тому кожен з розглянутих нижче популярних методів цифрового кодування має свої переваги й свої недоліки порівняно з іншими.

6.2 Цифрові коди

Потенційний код без повернення до нуля

На рис. 6.2, а показано уже згаданий раніше метод потенційного кодування, який також називається кодуванням без повернення до нуля (Non Return to Zero, NRZ). Остання назва відбиває ту обставину, що під час

передачі послідовності одиниць сигнал не вертається до нуля протягом такту (як ми побачимо далі, в інших методах кодування повернення до нуля в цьому випадку відбувається).

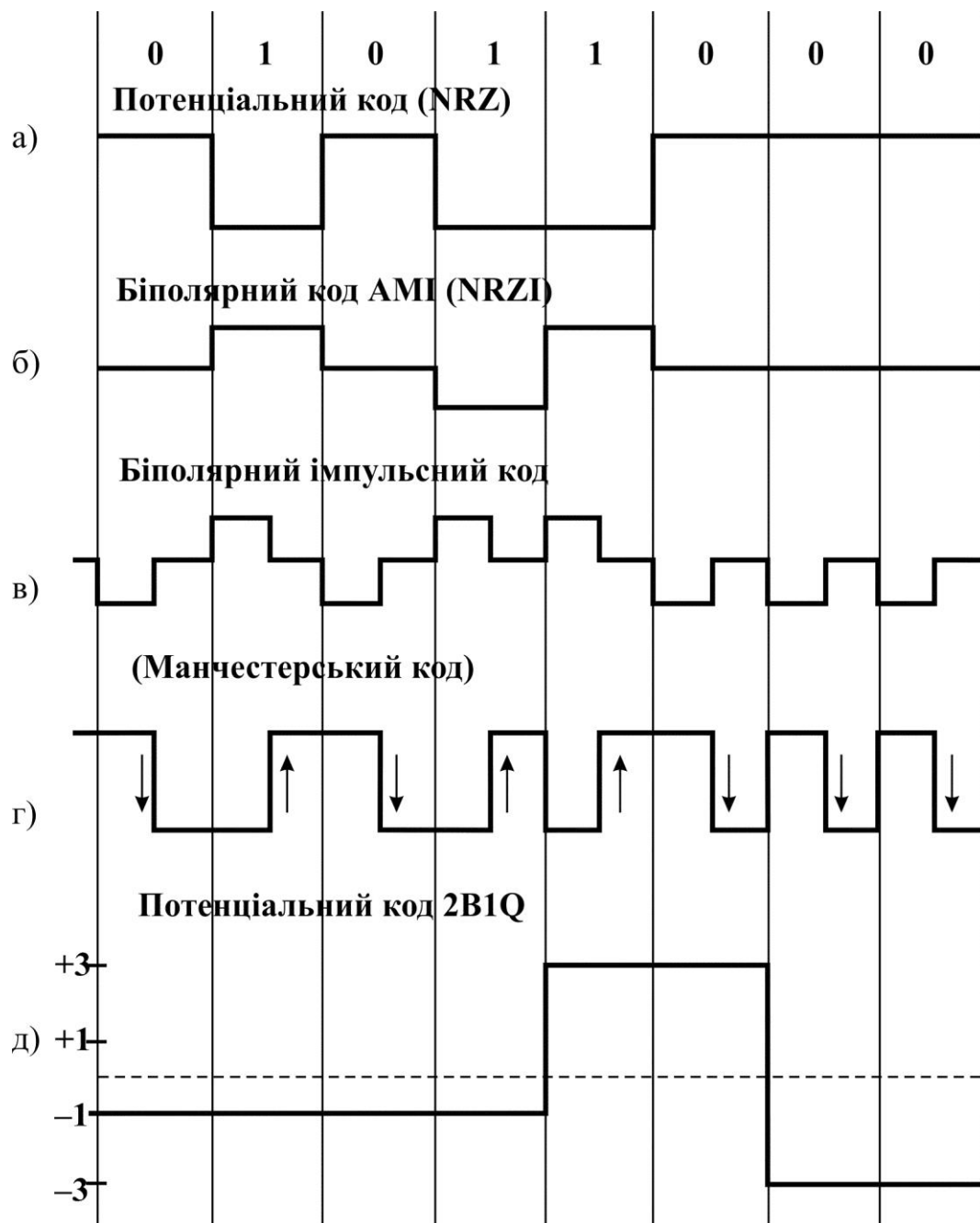


Рисунок 6.2 – Способи дискретного кодування даних

Метод NRZ простий в реалізації, має гарну розпізнаваність помилок (через два потенціали, які різко відрізняються), але йому не властива самосинхронізація. Під час передачі довгої послідовності одиниць або нулів сигнал на лінії не змінюється, тому приймач не має змоги визначати за вхідним сигналом моменти часу, коли слід наступного разу зчитувати дані. Навіть за наявності високоточного тактового генератора приймач може помилитися

з моментом знімання даних, оскільки частоти двох генераторів ніколи не бувають повністю ідентичними. Тому при високих швидкостях обміну даними й довгими послідовностями одиниць або нулів невелика неузгодженість тактових частот може призвести до помилки в цілий такт і, відповідно, зчитуванню некоректного значення біта.

Іншим серйозним недоліком методу NRZ є наявність низькочастотної складової, яка наближається до нуля під час передачі довгих послідовностей одиниць або нулів. Через це багато каналів зв'язку, які не забезпечують прямого гальванічного з'єднання між приймачем і джерелом, цей вид кодування не підтримують. У результаті в чистому вигляді код NRZ у мережах не використовується. Проте використовуються його різні модифікації, в яких усувають як погану самосинхронізацію коду NRZ, так і наявність постійної складової. Привабливість коду NRZ, через яку має сенс зайнятися його покращенням, полягає в досить низькій частоті основної гармоніки f_0 , яка дорівнює $N/2$ Гц, як це було показано в попередньому розділі. У інших методів кодування, наприклад манчестерського, основна гармоніка має вищу частоту.

Метод біполярного кодування з альтернативною інверсією

Однією з модифікацій методу NRZ є метод біполярного кодування з альтернативною інверсією (Bipolar Alternate Mark Inversion, AMI). У цьому методі (рис. 6.2, б) використовуються три рівні потенціалу – негативний, нульовий і позитивний. Для кодування логічного нуля використовується нульовий потенціал, а логічна одиниця кодується або позитивним потенціалом, або негативним, при цьому потенціал кожної нової одиниці протилежний потенціалу попередньої.

Код AMI частково ліквідує проблеми постійної складової й відсутності самосинхронізації, властиві коду NRZ. Це має місце під час передачі довгих послідовностей одиниць. У цих випадках сигнал на лінії є послідовністю різнополярних імпульсів з тим самим спектром, що й у коді NRZ, яка передає нулі і одиниці, тобто без постійної складової й з основною гармонікою $N/2$ Гц (де N – бітова швидкість передачі даних). Довгі ж послідовності нулів також небезпечні для коду AMI, як і для коду NRZ – сигнал вироджується в постійний потенціал нульової амплітуди. Тому код AMI потребує подальшого покращення, хоча задача спрощується – залишилося впоратися тільки з послідовностями нулів.

У цілому, для різних комбінацій біт на лінії використання коду АМІ призводить до більш вузького спектра сигналу, ніж для кода NRZ, а, значить, і до більш високої пропускної здатності лінії. Наприклад, при передачі одиниць і нулів, що чергуються, основна гармоніка f_0 має частоту $N/4$ Гц. Код АМІ надає також деякі можливості з розпізнавання помилкових сигналів. Так, порушення строгого чергування полярності сигналів говорить про помилковий імпульс або зникнення з лінії коректного імпульсу. Сигнал з некоректною полярністю називається забороненим сигналом (signal violation).

У коді АМІ використовуються не два, а три рівні сигнала на лінії. Додатковий рівень потребує збільшення потужності передавача приблизно на 3 дБ для забезпечення тієї самої вірогідності прийому біт на лінії, що є спільним недоліком кодів з декількома станами сигналу порівняно з кодами, які розрізняють тільки два стани.

Потенційний код з інверсією при одиниці

Існує код, схожий на АМІ, але тільки із двома рівнями сигналу. Під час передачі нуля він передає потенціал, що був установлений у попередньому такті (тобто не міняє його), а при передачі одиниці потенціал інвертується на протилежний. Цей код називається потенційним кодом з інверсією при одиниці (Non Return to Zero with ones Inverted, NRZI). Він зручний у тих випадках, коли використання третього рівня сигналу небажане, наприклад, в оптичних кабелях, де стійко розпізнаються два стани сигналу – одиниця й нуль.

Для покращення потенційних кодів, подібних АМІ і NRZI, використовуються два методи.

Перший метод заснований на додаванні у вихідний код надлишкових біт, які містять логічні одиниці. Очевидно, що в цьому випадку довгі послідовності нулів перериваються й код стає таким, що самосинхронізується, для будь-яких переданих даних. Зникає також постійна складова, а, значить, ще більше звужується спектр сигналу. Але цей метод знижує корисну пропускну здатність лінії, оскільки надлишкові одиниці користувальницької інформації не несуть.

Другий метод заснований на попередньому «перемішуванні» вихідної інформації так, щоб імовірність появи одиниць і нулів на лінії ставала близькою. Пристрої або блоки, що виконують таку операцію, називаються скремблерами (scramble – смітник, безладна збірка). Під час скремблювання використовується відомий алгоритм, тому приймач, отримавши двійкові дані, передає їх на дескремблер, який відновлює вихідну послідовність біт. Надлишкові біти

при цьому по лінії не передаються. Обидва методи належать до логічного, а не фізичного кодування, оскільки форму сигналів на лінії вони не визначають.

Біполярний імпульсний код

Крім потенційних кодів у мережах використовуються й імпульсні, коли дані подані повним імпульсом або ж його частиною – фронтом. Найпростішим випадком такого підходу є біполярний імпульсний код, у якому одиниця подана імпульсом однієї полярності, а нуль – іншої (рис. 6.2, в). Кожен імпульс триває половину такта. Такий код має чудові самосинхронізуючі властивості, але постійна складова може бути присутньою, наприклад, під час передачі довгої послідовності одиниць або нулів. Крім того, спектр у нього ширше, ніж у потенційних кодів. Так, під час передачі всіх нулів або одиниць частота основної гармоніки кода дорівнюватиме N Гц, що удвічі вище основної гармоніки коду NRZ і в чотири рази вище основної гармоніки коду АМІ при передачі одиниць і нулів, що чергуються. Через занадто широкий спектр біполярний імпульсний код використовується рідко.

Манчестерський код

У локальних мережах донедавна найпоширенішим методом кодування був так званий манчестерський код (рис. 6.2, г). Він застосовується в технологіях Ethernet і Token Ring.

У манчестерському коді для кодування одиниць і нулів використовується перепад потенціалу, тобто фронт імпульсу.

При манчестерському кодуванні кожен такт поділяється на дві частини. Інформація кодується перепадами потенціала, які відбуваються в середині кожного такту. Одиниця кодується перепадом від низького рівня сигналу до високого, а нуль – зворотним. На початку кожного такта може відбуватися службовий перепад сигналу, якщо потрібно подати кілька одиниць або нулів підряд. Оскільки сигнал змінюється принаймні один раз за такт передачі одного біта даних, то манчестерський код має гарні самосинхронізуючі властивості.

Смуга пропускання манчестерського коду вужче, ніж у біполярного імпульсного. У нього також немає постійної складової, а основна гармоніка в найгіршому разі (в ході передачі послідовності одиниць або нулів) має частоту N Гц, а в найкращому (в ході передачі одиниць і нулів, що чергуються) вона дорівнює $N/2$ Гц, як і у кодів АМІ або NRZ. У середньому

ширина смуги манчестерського кода у півтора рази вужча, ніж у біполярного імпульсного кода, а основна гармоніка коливається близько значення $3N/4$.

Манчестерський код має ще одну перевагу перед біполярним імпульсним кодом. В останньому для передачі даних використовуються три рівні сигналу, а в манчестерському – два.

Потенційний код 2B1Q

На рис. 6.2, д показаний потенційний код з чотирма рівнями сигналу для кодування даних. Це код 2B1Q, назва якого відбиває його суть – кожні два біти (2B) передаються за один такт сигналом, що має чотири стани (1Q). Парі біт 00 відповідає потенціал $-2,5$ В, парі біт 01 відповідає потенціал $-0,833$ В, парі 11 – потенціал $+0,833$ В, а парі 10 – потенціал $+2,5$ В. При цьому способі кодування потрібні додаткові заходи по боротьбі з довгими послідовностями однакових пар біт, оскільки при цьому сигнал перетворюється в постійну складову. При випадковому чергуванні біт спектр сигналу вдвічі вужчий, ніж у коді NRZ, оскільки при тій же бітовій швидкості тривалість такту збільшується вдвічі. Отже, за допомогою коду 2B1Q можна по одній тій самій лінії передавати дані удвічі швидше, ніж за допомогою коду AMI або NRZI. Однак для його реалізації потужність передавача має бути вище, щоб чотири рівні чітко розрізнялися приймачем на тлі завад.

Логічне кодування

Логічне кодування використовується для покращення потенційних кодів типу AMI, NRZI або 2Q1B. Логічне кодування має замінити довгі послідовності біт, що призводять до постійного потенціалу, вкрапленнями одиниць. Як ми вже відзначали вище, для логічного кодування характерні два методи – надлишкові коди й скремблювання.

Надлишкові коди

Надлишкові коди засновані на розбиванні вихідної послідовності біт на порції, які часто називають символами. Потім кожний вихідний символ замінюється на новий, який має більшу кількість біт, ніж вихідний. Наприклад, логічний код 4B/5B, що використовується у технологіях FDDI і Fast Ethernet, замінює вихідні символи довжиною в 4 біти на символи довжиною в 5 біт. Оскільки результуючі символи містять надлишкові біти, то загальна кількість бітових комбінацій у них більше, ніж у вихідних. Так, у коді 4B/5B результуючі символи можуть містити 32 бітові комбінації, у той час як вихідні символи – тільки 16. Тому в результуючому коді можна відібрати 16 таких комбінацій,

які не містять великої кількості нулів, а інші вважати забороненими кодами (code violation). Крім усунення постійної складової й додання коду властивості самосинхронізації, надлишкові коди дозволяють приймачу розпізнавати перекручені біти. Якщо приймач приймає заборонений код, значить, на лінії відбулося перекручування сигналу.

Відповідність вихідних і результуючих кодів 4В/5В подано у табл. 6.1.

Таблиця 6.1 – Відповідність вихідних і результуючих кодів 4В/5В

Вихідний код	Результуючий код	Вихідний код	Результуючий код
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

Код 4В/5В потім передається по лінії за допомогою фізичного кодування одним з методів потенційного кодування, сприйнятливому тільки до довгих послідовностей нулів. Символи коду 4В/5В довжиною 5 біт гарантують, що при будь-якому їхньому сполученні на лінії не можуть зустрітися більше трьох нулів підряд.

Буква В у назві коду означає, що елементарний сигнал має 2 стани – від англійського binary – двійковий. Є також коди й із трьома станами сигналу, наприклад, у коді 8В/6Т для кодування 8 біт вихідної інформації використовується код з 6 сигналів, кожен з яких має три стани. Надмірність коду 8В/6Т вище, ніж коду 4В/5В, оскільки на 256 вихідних кодів доводиться $3^6 = 729$ результуючих символів.

Використання таблиці перекодування є дуже простою операцією, тому цей підхід не ускладнює мережні адаптери й інтерфейсні блоки комутаторів та маршрутизаторів.

Для забезпечення заданої пропускної здатності лінії передавач, який використовує надлишковий код, має працювати з підвищеною тактовою частотою. Так, для передачі кодів 4В/5В зі швидкістю 100 Мб/с передавач має працювати з тактовою частотою 125 МГц. При цьому спектр сигналу на лінії розширюється порівняно з випадком, коли по лінії передається чистий, не надлишковий код. Проте, спектр надлишкового потенційного коду

виявляється вже спектром манчестерського коду, що виправдує додатковий етап логічного кодування, а також роботу приймача й передавача на підвищеній тактовій частоті.

Скремблювання

Перемішування даних скремблером перед передачею їх у лінію за допомогою потенційного коду є іншим способом логічного кодування.

Методи скремблювання полягають у побітному обчисленні результуючого коду на основі біт вихідного коду й отриманих у попередніх тактах біт результуючого коду. Наприклад, скремблер може реалізовувати таке співвідношення:

$$B_i = A_i \oplus B_{i-3} \oplus B_{i-5},$$

де B_i – двійкова цифра результуючого коду, отримана на i -му такті роботи скремблера;

A_i – двійкова цифра вихідного коду, що надходить на i -му такті на вхід скремблера;

B_{i-3} і B_{i-5} – двійкові цифри результуючого коду, отримані на попередніх тактах роботи скремблера, відповідно на 3 і 5 тактів раніше поточного такту;

$\oplus$ – операція, яка виключає АБО (додавання по модулю 2).

Наприклад, для вихідної послідовності 110110000001 скремблер дасть такий результуючий код:

$B_1 = A_1 = 1$ (перші три цифри результуючого коду будуть збігатися з вихідним, оскільки ще немає потрібних попередніх цифр)

$$B_2 = A_2 = 1$$

$$B_3 = A_3 = 0$$

$$B_4 = A_4 \oplus B_1 = 1 \oplus 1 = 0$$

$$B_5 = A_5 \oplus B_2 = 1 \oplus 1 = 0$$

$$B_6 = A_6 \oplus B_3 \oplus B_1 = 0 \oplus 0 \oplus 1 = 1$$

$$B_7 = A_7 \oplus B_4 \oplus B_2 = 0 \oplus 0 \oplus 1 = 1$$

$$B_8 = A_8 \oplus B_5 \oplus B_3 = 0 \oplus 0 \oplus 0 = 0$$

$$B_9 = A_9 \oplus B_6 \oplus B_4 = 0 \oplus 1 \oplus 0 = 1$$

$$B_{10} = A_{10} \oplus B_7 \oplus B_5 = 0 \oplus 1 \oplus 0 = 1$$

$$B_{11} = A_{11} \oplus B_8 \oplus B_6 = 0 \oplus 0 \oplus 1 = 1$$

$$B_{12} = A_{12} \oplus B_9 \oplus B_7 = 1 \oplus 1 \oplus 1 = 1$$

Отже, на виході скремблера з'явиться послідовність 110001101111, у якій немає послідовності із шести нулів, що була присутньою у вихідному коді.

Після отримання результуючої послідовності приймач передає її дескремблеру, який відновлює вихідну послідовність на основі зворотного співвідношення:

$$C_i = B_i \oplus B_{i-3} \oplus B_{i-5} = (A_i \oplus B_{i-3} \oplus B_{i-5}) \oplus B_{i-3} \oplus B_{i-5} = A_i.$$

Різні алгоритми скремблювання відрізняються кількістю доданків, що дають цифру результуючого коду, і зрушенням між доданками. Так, у мережах ISDN під час передачі даних від мережі до абонента використовується перетворення зі зрушеннями в 5 і 23 позиції, а в ході передачі даних від абонента в мережу – зі зрушеннями 18 і 23 позиції.

Існують і простіші методи боротьби з послідовностями одиниць, які також належать до класу скремблювання.

Для покращення коду Bipolar AMI використовуються два методи, засновані на штучному перекручуванні послідовності нулів забороненими символами.

На рис. 6.3 показано використання методу B8ZS (Bipolar with 8-Zeros Substitution) і методу HDB3 (High-Density Bipolar 3-Zeros) для корегування коду AMI. Вихідний код складається з двох довгих послідовностей нулів: у першому випадку – з 8, а в другому – з 5.

Код B8ZS виправляє тільки послідовності, які складаються з 8 нулів. Для цього він після перших трьох нулів замість п'яти нулів, що залишилися, вставляє п'ять цифр: $\vee - 1^* - 0 - \vee - 1^*$. Знак « $\vee$ » тут позначає сигнал одиниці, забороненої для даного такту полярності, тобто сигнал, який не змінює полярність попередньої одиниці, 1^* – сигнал одиниці коректної полярності, а знак зірочки зазначає той факт, що у вихідному коді в цьому такті була не одиниця, а нуль. У результаті на восьми тактах приймач спостерігає два перекручування – дуже мало ймовірно, що це сталося через шум на лінії або інших збоїв передачі. Тому приймач вважає такі порушення кодуванням 8 послідовних нулів і після прийому заміняє їх на вихідні 8 нулів. Код B8ZS побудований так, що його постійна складова дорівнює нулю за будь-яких послідовностей двійкових цифр.

Код HDB3 виправляє будь-які чотири нулі, що йдуть підряд, у вихідній послідовності. Правила формування коду HDB3 складніші, ніж коди B8ZS. Кожні чотири нулі замінюються чотирма сигналами, у яких є один сигнал $\vee$. Для придушення постійної складової полярності сигналу $\vee$ чергується під час послідовних заміन. Крім того, для заміни використовуються два зразки чотиритактових кодів. Якщо перед заміною вихідний код містив непарну кількість одиниць, то використовується послідовність $000\vee$, а якщо кількість одиниць була парною – послідовність $1^*00\vee$.

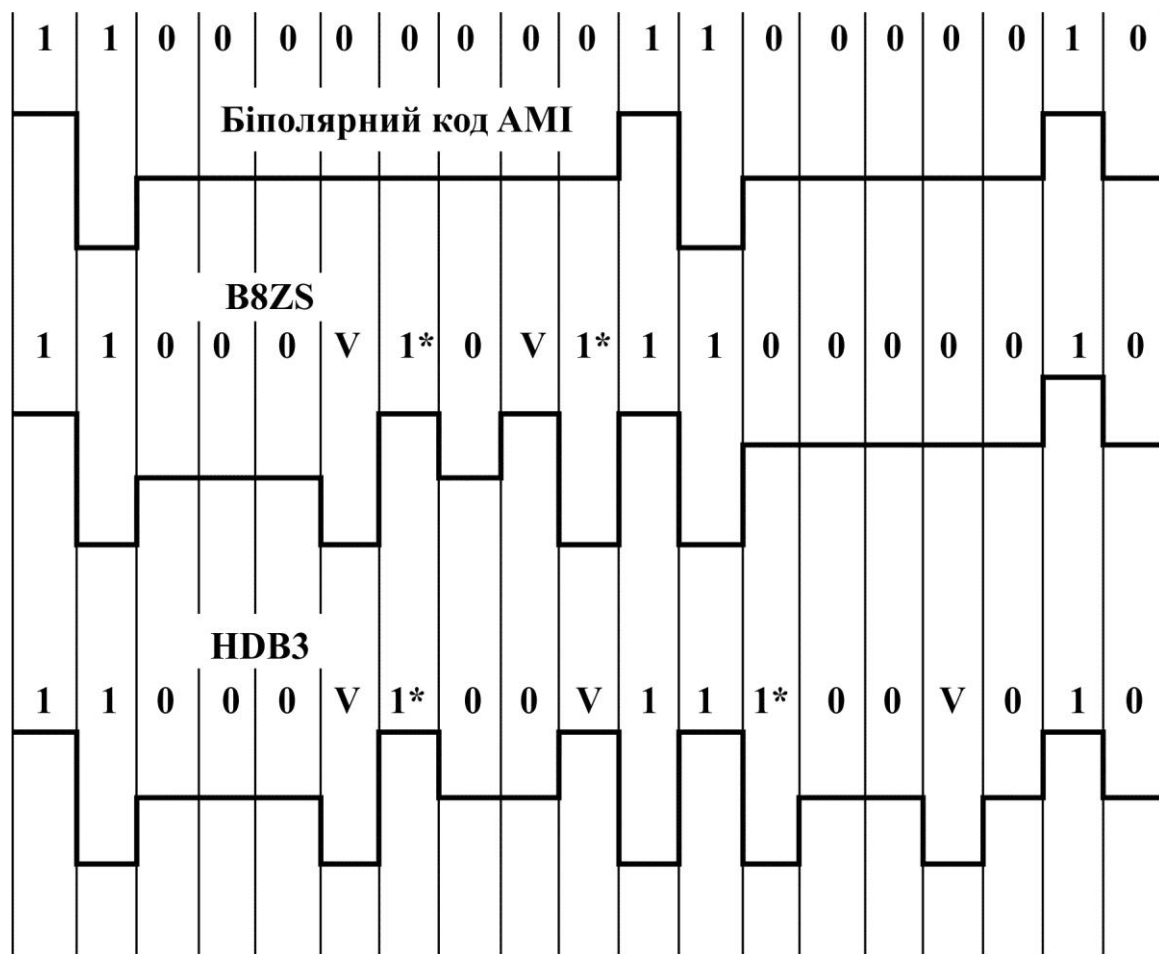


Рисунок 6.3 – Коди B8ZS і HDB3:

✓ – сигнал одиниці забороненої полярності; 1* – сигнал одиниці коректної полярності, але такої, що замінила 0 у вихідному коді

Покращені потенційні коди мають досить вузьку смугу пропускання для будь-яких послідовностей одиниць і нулів, що зустрічаються в переданих даних. На рис. 6.4 наведено спектри сигналів різних кодів, отримані в ході передачі довільних даних, у яких різні сполучення нулів і одиниць у вихідному коді рівновірогідні. В ході побудови графіків спектр усереднювався за всіма можливими наборами вихідних послідовностей. Природно, що результуючі коди можуть мати й інший розподіл нулів і одиниць. З рис. 6.4 видно, що потенційний код NRZ має гарний спектр з одним недоліком – у нього є постійна складова. Коди, отримані з потенційного шляхом логічного кодування, мають більш вузький спектр, ніж манчестерський, навіть при підвищеній тактовій частоті (на рисунку спектр коду 4B/5B мав би приблизно збігатися з кодом B8ZS, але він зсунутий в область більш високих частот, оскільки його тактова частота підвищена на 1/4 порівняно з іншими кодами). Цим пояснюється застосування потенційних надлишкових і скремблених

кодів у сучасних технологіях, подібних до FDDI, Fast Ethernet, Gigabit Ethernet, ISDN тощо, замість манчестерського й біполярного імпульсного кодування.



Рисунок 6.4 – Спектри потенційних і імпульсних кодів

Дискретна модуляція аналогових сигналів

Однією з основних тенденцій розвитку мережних технологій є передача в одній мережі як дискретних, так і аналогових за своєю природою даних. Джерелами дискретних даних є комп'ютери та інші обчислювальні пристрої, а джерелами аналогових даних є такі пристрої, як телефони, відеокамери, звуко- і відеовідтворююча апаратура. На ранніх етапах вирішення цієї проблеми в територіальних мережах усі типи даних передавалися в аналоговій формі, при цьому дискретні за своїм характером комп'ютерні дані перетворювалися в аналогову форму за допомогою модемів.

Однак з розвитком техніки знімання й передачі аналогових даних з'ясувалося, що передача їх в аналоговій формі не дозволяє покращити якість прийнятих на другому кінці лінії даних, якщо вони значно викривилися під час передачі. Сам аналоговий сигнал не дає ніяких зазначень ні про те, що відбулося перекручування, ні про те, як його виправити, оскільки форма сигналу може бути будь-якою, у тому числі й такою, яку зафіксував приймач. Покращення ж якості ліній, особливо територіальних, вимагає величезних зусиль і капіталовкладень. Тому на зміну аналоговій техніці запису й передачі звуку й зображення прийшла цифрова. Ця техніка використовує так звану дискретну модуляцію вихідних безперервних у часі аналогових процесів.

Дискретні способи модуляції засновані на дискретизації безперервних процесів як по амплітуді, так і за часом (рис. 6.5). Розглянемо принципи дискретної модуляції на прикладі імпульсно-кодової модуляції, ІКМ (Pulse Amplitude Modulation, PAM), яка широко застосовується в цифровій телефонії.

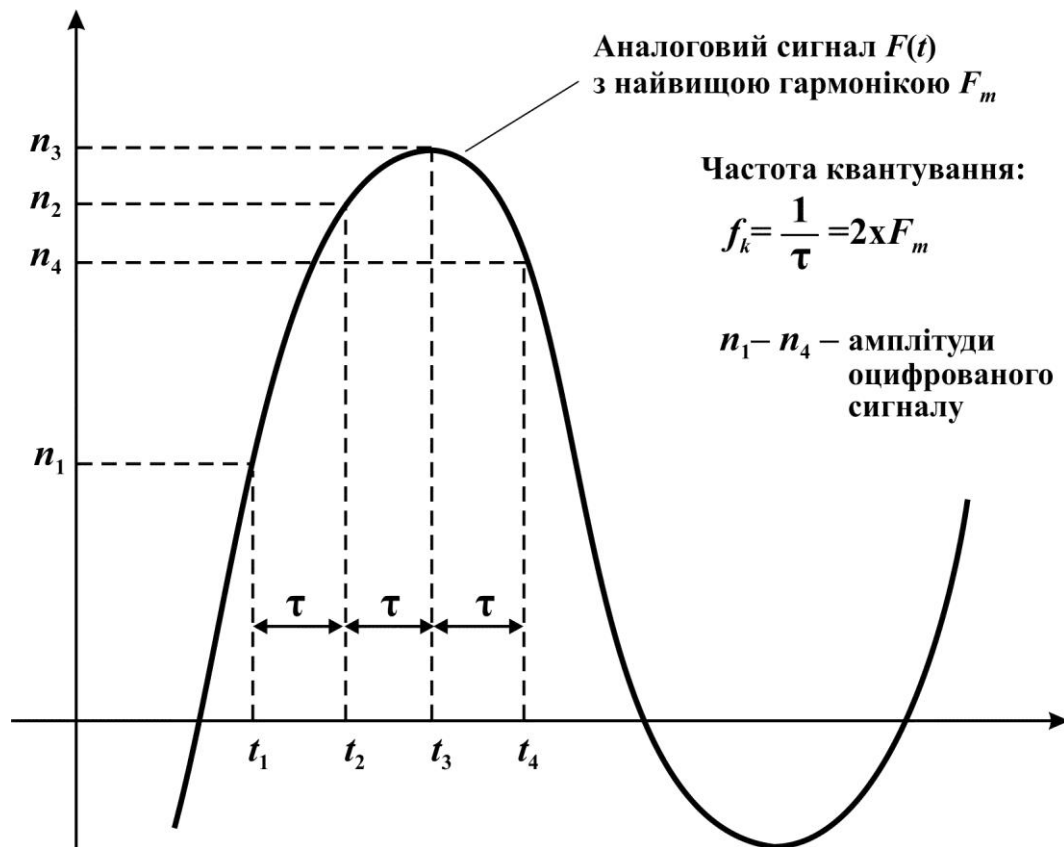


Рисунок 6.5 – Дискретна модуляція безперервного процесу

Амплітуда вихідної безперервної функції вимірюється із заданим періодом – за рахунок цього відбувається дискретизація за часом. Потім кожен вимір подається у вигляді двійкового числа певної розрядності, що означає дискретизацію за значеннями функції – безперервна множина можливих значень амплітуди заміняється дискретною множиною її значень. Пристрій, який виконує подібну функцію, називається аналого-цифровим перетворювачем (АЦП). Після цього виміри передаються по каналах зв'язку у вигляді послідовності одиниць і нулів. При цьому застосовуються ті самі методи кодування, що й у випадку передачі від початку дискретної інформації, тобто, наприклад, методи, засновані на коді B8ZS або 2B1Q.

На приймальній стороні лінії коди перетворюються у вихідну послідовність біт, а спеціальна апаратура, яка називається цифро-аналоговим перетворювачем (ЦАП), проводить демодуляцію оцифрованих амплітуд безперервного сигналу, відновлюючи вихідну безперервну функцію часу.

Дискретна модуляція заснована на теорії відображення Найквіста – Котельникова. Відповідно до неї, аналогова безперервна функція, передана у вигляді послідовності її дискретних за часом значень, може бути точно відновлена, якщо частота дискретизації була удвічі або більше разів вища, ніж частота найвищої гармоніки спектра вихідної функції.

Якщо ця умова не дотримується, то відновлена функція буде суттєво відрізнятися від вихідної.

Перевагою цифрових методів запису, відтворення й передачі аналогової інформації є можливість контролю вірогідності зчитаних з носія або отриманих по лінії зв'язку даних. Для цього можна використовувати ті самі методи, які застосовуються для комп'ютерних даних (і розглядаються докладніше далі), – обчислення контрольної суми, повторна передача перекручених кадрів, застосування кодів, що самокорегуються.

Для якісної передачі голосу в методі ІКМ використовується частота квантування амплітуди звукових коливань в 8000 Гц. Це пов'язано з тим, що в аналоговій телефонії для передачі голосу був обраний діапазон від 300 до 3400 Гц, який досить якісно передає всі основні гармоніки співрозмовників. Відповідно до теореми Найквіста – Котельникова для якісної передачі голосу досить обрати частоту дискретизації, яка удвічі перевищує найвищу гармоніку безперервного сигналу, тобто $2 \times 3400 = 6800$ Гц. В дійсності обрана частота дискретизації 8000 Гц забезпечує деякий запас якості. У методі ІКМ зазвичай використовується 7 або 8 біт коду для подання амплітуди одного виміру. Відповідно, це дає 127 або 256 градацій звукового сигналу, що виявляється цілком достатнім для якісної передачі голосу. Під час використання методу ІКМ для передачі одного голосового каналу необхідна пропускна здатність 56 або 64 Кбіт/с залежно від того, якою кількістю біт подається кожний вимір. Якщо для цих цілей використовується 7 біт, то при частоті передачі вимірів в 8000 Гц отримаємо: $8000 \times 7 = 56000$ біт/с або 56 Кбіт/с; а для випадку 8 біт: $8000 \times 8 = 64000$ біт/с або 64 Кбіт/с.

Стандартним є цифровий канал 64 Кбіт/с, який також називається елементарним каналом цифрових телефонних мереж.

Передача безперервного сигналу в дискретному вигляді вимагає від мереж чіткого дотримання часового інтервалу в 125 мкс (що відповідає частоті дискретизації 8000 Гц) між сусідніми вимірами, тобто вимагає синхронної передачі даних між вузлами мережі. При недотриманні синхронності вимірів, які прибувають, вихідний сигнал відновлюється неправильно, що призводить до перекручування голосу, зображення або іншої мультимедійної інформації,

переданої по цифрових мережах. Так, перекручування синхронізації може призвести до ефекту «луни», а зрушення між вимірами в 200 мс призводять до втрати розпізнаваності вимовних слів. Водночас втрата одного виміру при дотриманні синхронності між іншими вимірами практично не позначається на відтвореному звуці. Це відбувається за рахунок тих пристроїв, що згладжують, в цифро-аналогових перетворювачах, які засновані на властивості інерційності будь-якого фізичного сигналу – амплітуда звукових коливань не може миттєво змінитися на велику величину. На якість сигналу після ЦАП впливає не тільки синхронність надходження на його вхід вимірів, але й похибка дискретизації амплітуд цих вимірів. У теоремі Найквіста – Котельникова передбачається, що амплітуди функції вимірюються точно, водночас використання для їхнього зберігання двійкових чисел з обмеженою розрядністю трохи спотворює ці амплітуди. Відповідно спотворюється відновлений безперервний сигнал, який називається шумом дискретизації (по амплітуді).

Існують й інші методи дискретної модуляції, які дозволяють подавати виміри голосу в компактнішій формі, наприклад, у вигляді послідовності 4-бітних або 2-бітних чисел. При цьому один голосовий канал вимагає меншої пропускної здатності, наприклад, 32 Кбіт/с, 16 Кбіт/с або ще менше. З 1985 року застосовується стандарт CCITT кодування голосу, яке називається Adaptive Differential Pulse Code Modulation (ADPCM). Коди ADPCM засновані на знаходженні різниць між послідовними вимірами голосу, які потім і передаються мережею. У коді ADPCM для зберігання однієї різниці використовуються 4 біт і голос передається зі швидкістю 32 Кбіт/с. Сучасніший метод, Linear Predictive Coding (LPC), робить виміри вихідної функції рідше, але використовує методи прогнозування напряму зміни амплітуди сигналу. За допомогою цього методу можна понизити швидкість передачі голосу до 9600 біт/с.

Подані у цифровій формі безперервні дані легко можна передати через комп'ютерну мережу. Для цього досить помістити кілька вимірів у кадр будь-якої стандартної мережної технології, забезпечити кадр правильною адресою призначення й відправити адресату. Адресат має витягти з кадра виміри й подати їх з частотою квантування (для голосу – із частотою 8000 Гц) на цифро-аналоговий перетворювач. З надходженням наступних кадрів з вимірами голосу операція має повторитися. Якщо кадри прибуватимуть досить синхронно, то якість голосу може бути достатньо високою. Однак, як ми вже знаємо, кадри в комп'ютерних мережах можуть затримуватися як у кінцевих вузлах (при очікуванні доступу до поділюваного середовища), так і в проміжних

комунікаційних пристроях – мостах, комутаторах і маршрутизаторах. Тому якість голосу при передачі в цифровій формі через комп'ютерні мережі зазвичай буває невисокою. Для якісної передачі оцифрованих безперервних сигналів – голосу, зображення – сьогодні використовують спеціальні цифрові мережі, такі як ISDN, ATM, і мережі цифрового телебачення. Проте, для передачі внутрішньокорпоративних телефонних розмов сьогодні характерні мережі frame relay, затримки передачі кадрів яких укладаються в припустимі межі.

Асинхронна й синхронна передачі

Під час обміну даними на фізичному рівні одиницею інформації є біт, тому засоби фізичного рівня завжди підтримують побітову синхронізацію між приймачем і передавачем.

Канальний рівень оперує кадрами даних і забезпечує синхронізацію між приймачем і передавачем на рівні кадрів. В обов'язки приймача входить розпізнавання початку першого байта кадра, розпізнавання меж полів кадра й розпізнавання ознаки закінчення кадра.

Зазвичай достатньо забезпечити синхронізацію на зазначених двох рівнях – бітовому й кадровому, – щоб передавач і приймач змогли забезпечити стійкий обмін інформацією. Однак при поганій якості лінії зв'язку (зазвичай це відноситься до телефонних комутованих каналів) для здешевлення апаратури й підвищення надійності передачі даних вводять додаткові засоби синхронізації на рівні байт.

Такий режим роботи називається асинхронним або старт-стопним. Іншою причиною використання такого режиму роботи є наявність пристроїв, які генерують байти даних у випадкові моменти часу. Так працює клавіатура дисплея або іншого термінального пристрою, з якого людина вводить дані для обробки їх комп'ютером.

В асинхронному режимі кожен байт даних супроводжується спеціальними сигналами «старт» і «стіп» (рис. 6.6, а). Призначення цих сигналів у тому, щоб, по-перше, сповістити приймач про прихід даних і, по-друге, щоб дати приймачу достатньо часу для виконання деяких функцій, пов'язаних із синхронізацією, до надходження наступного байта. Сигнал «старт» має тривалість в один тактовий інтервал, а сигнал «стіп» може тривати один, півтора або два такти, тому говорять, що використовується один, півтора або два біти як стоповий сигнал, хоча користувальницькі біти ці сигнали не подають.

Асинхронним описаний режим називається тому, що кожен байт може бути трохи зміщений у часі щодо побітових тактів попереднього байта.

Така асинхронність передачі байт не впливає на коректність прийнятих даних, оскільки на початку кожного байта відбувається додаткова синхронізація приймача з джерелом завдяки бітам «старт». Більш «вільні» часові допуски визначають низьку вартість обладнання асинхронної системи.

При синхронному режимі передачі старт-стопні біти між кожною парою байт відсутні. Користувальницькі дані збираються в кадр, який випереджається байтами синхронізації (рис. 6.6, б). Байт синхронізації – це байт, який містить заздалегідь відомий код, наприклад 0111110, який сповіщає приймач про прихід кадру даних. При його отриманні приймач має увійти в байтовий синхронізм із передавачем, тобто правильно розуміти початок чергового байта кадру. Іноді застосовується декілька синхробайт для забезпечення надійнішої синхронізації приймача й передавача. Оскільки при передачі довгого кадру у приймача можуть з'явитися проблеми з синхронізацією біт, то в цьому випадку використовуються коди, які самосинхронізуються.

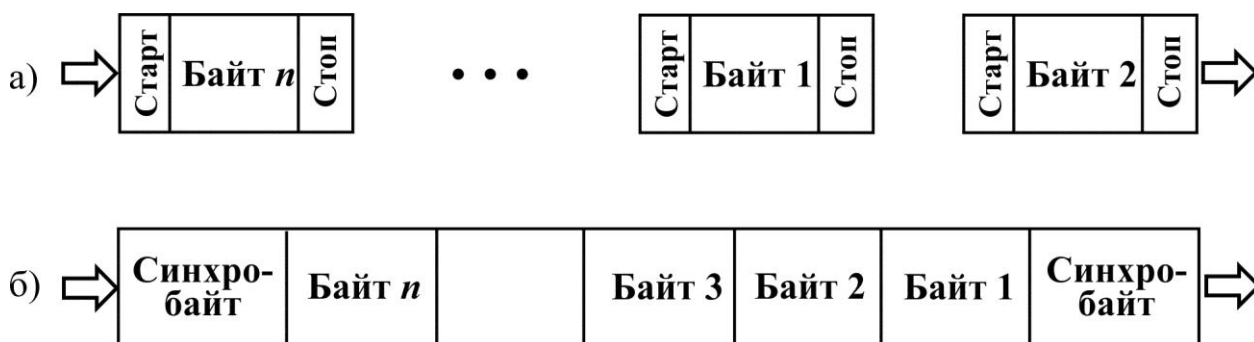


Рисунок 6.6 – Асинхронна (а) і синхронна (б) передачі на рівні байт

Висновки

1. Під час передачі дискретних даних по вузькополосному каналу тональної частоти, який використовується в телефонії, найбільш підходящими виявляються способи аналогової модуляції, при яких несуча синусоїда модулюється вихідною послідовністю двійкових цифр. Ця операція здійснюється спеціальними пристроями – модемами.

2. Для низькошвидкісної передачі даних застосовується зміна частоти несучої синусоїди. Більш високошвидкісні модеми працюють на комбінованих способах квадратурної амплітудної модуляції (QAM), для якої характерні 4 рівні амплітуди несучої синусоїди й 8 рівнів фази. Не всі з можливих 32 комбінацій методу QAM використовуються для передачі даних, заборонені комбінації дозволяють розпізнавати перекручені дані на фізичному рівні.

3. На широкополосних каналах зв'язку застосовуються потенційні й імпульсні методи кодування, у яких дані подані різними рівнями постійного потенціалу сигналу або полярностями імпульсу чи його фронту.

4. Під час використання потенційних кодів особливого значення набуває задача синхронізації приймача з передавачем, оскільки під час передачі довгих послідовностей нулів або одиниць сигнал на вході приймача не змінюється й приймачу складно визначити момент знімання чергового біта даних.

5. Найбільш простим потенційним кодом є код без повернення до нуля (NRZ), однак він не той, що самосинхронізується, й створює постійну складову.

6. Найбільш популярним імпульсним кодом є манчестерський код, у якому інформацію несе напрям перепаду сигналу в середині кожного такта. Манчестерський код застосовується в технологіях Ethernet і Token Ring.

7. Для покращення властивостей потенційного коду NRZ використовуються методи логічного кодування, які виключають довгі послідовності нулів. Ці методи засновані на:

- введенні надлишкових біт у вихідні дані (коди типу 4B/5B);
- скремблюванні вихідних даних (коди типу 2B1Q).

8. Покращені потенційні коди мають вузький спектр, ніж імпульсні, тому вони знаходять застосування у високошвидкісних технологіях, таких, як FDDI, Fast Ethernet, Gigabit Ethernet.

6.3 Контрольні запитання та завдання

1. Поясніть, з яких міркувань обрана пропускна здатність 64 Кбіт/с елементарного каналу цифрових телефонних мереж?

2. Чим визначається побудова високошвидкісних модемів на комбінованих схемах із квадратурною амплітудною модуляцією сигналів (QAM)?

3. Назвіть методи компресії, які найкраще підходять для текстової інформації. Чому вони неефективні для стиснення двійкових даних?

4. Запропонуйте коди нерівної довжини для кожного із символів A, B, C, D, F і O, якщо потрібно передати повідомлення BDDACAAFOOOAOOOO. Чи буде досягнута компресія даних порівняно з використанням:

- традиційних кодів ASCII?
- кодів однакової довжини, які враховують наявність тільки даних символів?

5. Які цифрові коди застосовуються в мережах FDDI, Fast Ethernet, Gigabit Ethernet?

Лекція 7

МЕТОДИ КАДРУВАННЯ

7.1 Знакове кадрювання

Методи передачі даних канального рівня

Канальний рівень забезпечує передачу пакетів даних, що надходять від протоколів верхніх рівнів, вузлу призначення, адреса якого також вказує протокол верхнього рівня. Протоколи канального рівня оформляють передані їм пакети в кадри власного формату, поміщаючи зазначену адресу призначення в одне з полів такого кадра, а також супроводжуючи кадр контрольною сумою. Протокол канального рівня має локальне значення, він призначений для доставки кадрів даних, як правило, у межах мереж з простою топологією зв'язків і однотипною або близькою технологією, наприклад, в односегментних мережах Ethernet або ж у багатосегментних мережах Ethernet і Token Ring ієрархічної топології, розділених тільки мостами й комутаторами. В усіх цих конфігураціях адреса призначення має локальний зміст для даної мережі й не змінюється під час проходження кадра від вузла-джерела до вузла-призначення. Можливість передавати дані між локальними мережами різних технологій пов'язана з тим, що в цих технологіях використовуються адреси однакового формату, до того ж виробники мережних адаптерів забезпечують унікальність адрес незалежно від технології.

Іншою галуззю дії протоколів канального рівня є зв'язки типу «точка-точка» глобальних мереж, коли протокол канального рівня відповідальний за доставку кадра безпосередньому сусіду. Адреса в цьому випадку не має принципового значення, а на перший план виходить здатність протоколу відновлювати перекручені й загублені кадри, оскільки погана якість територіальних каналів, особливо комутованих телефонних, часто вимагає виконання подібних дій.

Якщо ж перераховані вище умови не дотримуються, наприклад, зв'язки між сегментами Ethernet мають петлевидну структуру, або поєднувані мережі використовують різні способи адресації, як це має місце в мережах Ethernet і X.25, то протокол канального рівня не може самотужки впоратися з задачею передачі кадра між вузлами й вимагає допомоги протокола мережного рівня.

Найістотнішими характеристиками методу передачі, а, значить, і протоколу, що працює на каналному рівні, є такі:

- асинхронний/синхронний;
- символно-орієнтований/біт-орієнтований;
- з попереднім встановленням з'єднання/дейтаграмний;
- з виявленням перекручених даних/без виявлення;
- з виявленням загублених даних/без виявлення;
- з відновленням перекручених і загублених даних/без відновлення;
- з підтримкою динамічної компресії даних/без підтримки.

Багато із цих властивостей характерні не тільки для протоколів каналного рівня, але й для протоколів більш високих рівнів.

Асинхронні протоколи

Асинхронні протоколи є найстарішим способом зв'язку. Ці протоколи оперують не з кадрами, а з окремими символами, які подані байтами зі старт-стоповими символами. Асинхронні протоколи походять від тих часів, коли дві людини зв'язувалися за допомогою телетайпів каналом «точка-точка». З розвитком техніки асинхронні протоколи стали застосовуватися для зв'язку телетайпів, різного роду клавіатур і дисплеїв з обчислювальними машинами. Одиницею переданих даних був не кадр даних, а окремий символ. Деякі символи мали керуючий характер, наприклад, символ <CR> пропонував телетайпу або дисплею виконати повернення каретки на початок рядка. У цих протоколах існують керуючі послідовності, які зазвичай починаються із символу <ESC>. Ці послідовності викликали на керованому пристрої досить складні дії, – наприклад, завантаження нового шрифту в принтер.

В асинхронних протоколах застосовуються стандартні набори символів, найчастіше ASCII або EBCDIC. Оскільки перші 32 або 27 кодів у цих наборах є спеціальними кодами, які не відображаються на дисплеї або принтері, то вони використовувалися асинхронними протоколами для управління режимом обміну даними. У самих користувальницьких даних, які являли собою літери, цифри, а також такі знаки, як @, %, \$ тощо, спеціальні символи ніколи не зустрічалися, так що проблеми їхнього відділення від користувальницьких даних не існувало.

Поступово асинхронні протоколи ускладнювалися й стали поряд з окремими символами використовувати цілі блоки даних, тобто кадри. Наприклад, популярний протокол XMODEM передає файли між двома комп'ютерами асинхронним модемом. Початок прийому чергового блока файлу

ініціюється символьною командою – приймаюча сторона постійно передає символ ASCII NAK. Передавальна сторона, прийнявши NAK, відправляє черговий блок файлу, який складає з 128 байт даних, заголовок й кінцевика. Заголовок складається зі спеціального символу SOH (Start Of Header) і номера блока. Кінцевик містить контрольну суму блока даних. Приймальна сторона, отримавши новий блок, перевіряла його номер і контрольну суму. У випадку збігу цих параметрів з очікуваними приймач відправляв символ ACK, а в протилежному випадку – символ NAK, після чого передавач мав повторити передачу даного блока. Наприкінці передачі файлу передавався символ EOX.

Як видно з опису протоколу XMODEM, частина управляючих операцій виконувалася в асинхронних протоколах посиленням в асинхронному режимі окремих символів, у той же час частина даних пересилалася блоками, що характерніше для синхронних протоколів.

Синхронні символьно-орієнтовані й біт-орієнтовані протоколи

У синхронних протоколах між символами, що пересилаються (байтами), немає стартових і стопових сигналів, тому окремі символи в цих протоколах пересилати не можна. Всі обміни даними здійснюються кадрами, які мають у загальному випадку заголовок, поле даних і кінцевик (рис. 7.1). Усі біти кадра передаються безперервним синхронним потоком, що значно прискорює передачу даних.

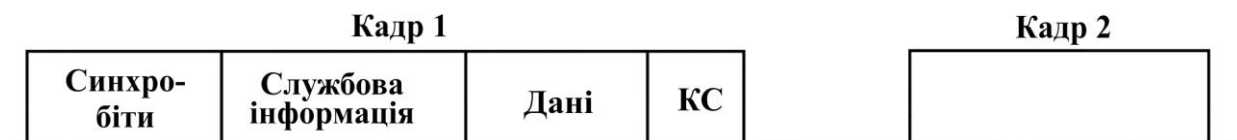


Рисунок 7.1 – Кадри синхронних протоколів

Оскільки байти в цих протоколах не відділяються один від одного службовими сигналами, то однією з перших задач приймача є розпізнавання межі байт. Потім приймач має знайти початок і кінець кадра, а також визначити межі кожного поля кадра – адреси призначення, адреси джерела, інших службових полів заголовка, поля даних і контрольної суми, якщо вона є.

Більшість протоколів допускає використання в кадрі поля даних змінної довжини. Іноді й заголовок може мати змінну довжину. Зазвичай протоколи визначають максимальне значення, яке може мати довжина поля даних. Ця величина називається максимальною одиницею передачі даних (Maximum Transfer Unit, MTU). У деяких протоколах задається також мінімальне значення, яке може мати довжина поля даних. Наприклад, протокол Ethernet

вимагає, щоб поле даних містило, принаймні, 46 байт даних (якщо додаток хоче відправити меншу кількість байт, то він зобов'язаний доповнити їх до 46 байт будь-якими значеннями). Інші протоколи дозволяють використовувати поле даних нульової довжини, наприклад FDDI.

Існують також протоколи з кадрами фіксованої довжини, наприклад, у протоколі АТМ кадри фіксованого розміру 53 байт, включаючи службову інформацію. Для таких протоколів необхідно вирішити тільки першу частину задачі – розпізнати початок кадра.

Синхронні протоколи каналного рівня бувають двох типів: символно-орієнтовані (байт-орієнтовані) і біт-орієнтовані. Для обох характерні одні й ті самі методи синхронізації біт. Головне розходження між ними полягає в методі синхронізації символів і кадрів.

Символьно-орієнтовані протоколи

Символьно-орієнтовані протоколи використовуються, в основному, для передачі блоків відображуваних символів, наприклад, текстових файлів. Оскільки під час синхронної передачі немає стопових і стартових біт, для синхронізації символів необхідний інший метод. Синхронізація досягається за рахунок того, що передавач додає два або більше керуючих символів, які називаються символами SYN, перед кожним блоком символів. У коді ASCII символ SYN має двійкове значення 0010110, це несиметричне відносно початку символа значення дозволяє легко розмежовувати окремі символи SYN під час їхнього послідовного прийому. Символи SYN виконують дві функції. По-перше, вони забезпечують приймачу побітову синхронізацію; по-друге, як тільки бітова синхронізація досягається, вони дозволяють приймачу почати розпізнавання меж символів SYN. Після того як приймач почав відокремлювати один символ від другого, можна задавати межі початку кадра за допомогою іншого спеціального символу. Зазвичай в символних протоколах для цих цілей використовується символ STX (Start of TeXt, ASCII 0000010). Інший символ відзначає закінчення кадра – ETX (End of TeXt, ASCII 0000011).

Однак такий простий спосіб виділення початка й кінця кадра добре працював тільки в тому випадку, якщо усередині кадра не було символів STX і ETX. При підключенні до комп'ютера алфавітно-цифрових терміналів така задача дійсно не виникала. Проте, синхронні символно-орієнтовані протоколи пізніше стали використовуватися й для зв'язку комп'ютера з комп'ютером, а в цьому випадку дані всередині кадра можуть бути будь-які, якщо, наприклад, між комп'ютерами передається програма. Найбільш популярним протоколом

такого типу був протокол BSC компанії IBM. Він працював у двох режимах – непрозорому, у якому деякі спеціальні символи всередині кадра заборонялися, і прозорому, у якому дозволялася передача всередині кадра будь-яких символів, у тому числі й ETX. Прозорість досягалася за рахунок того, що перед управляючими символами STX і ETX завжди вставлявся символ DLE (Data Link Escape). Така процедура називається стаффінгом символів (stuff – всяка всячина, заповнювач). А якщо в полі даних кадра зустрічалася послідовність DLE ETX, то передавач подвоював символ DLE, тобто породжував послідовність DLE DLE ETX. Приймач, зустрівши підряд два символи DLE DLE, завжди видаляв перший, але той DLE, що залишився, уже не розглядав як початок управляючої послідовності, тобто символи, що залишилися, DLE ETX вважав просто користувальницькими даними.

7.2 Бітове кадрування

Біт-орієнтовані протоколи

Потреба в парі символів на початку й у кінці кожного кадра разом з додатковими символами DLE означає, що символно-орієнтована передача не ефективна для передачі двійкових даних, оскільки доводиться в поле даних кадра додавати досить багато надлишкових даних. Крім того, формат управляючих символів для різних кодувань різних, наприклад, у кодї ASCII символ SYN дорівнює 0010110, а в кодї EBCDIC – 00110010. Так що цей метод допустимий тільки з певним типом кодування, навіть якщо кадр містить виключно двійкові дані. Щоб перебороти ці проблеми, на сьогодні майже завжди використовується більш універсальний метод, який називається біт-орієнтованою передачею. Цей метод тепер застосовується в ході передачі як двійкових, так і символних даних.

На рис. 7.2 показані три різні схеми біт-орієнтованої передачі. Вони відрізняються способом позначення початку й кінця кожного кадра.

Перша схема, показана на рис. 7.2, а, схожа на схему із символами STX і ETX у символно-орієнтованих протоколах. Початок і кінець кожного кадра позначається однією й тією самою 8-бітовою послідовністю – 01111110, яка називається прапорцем. Термін «біт-орієнтований» використовується, оскільки прийнятий потік біт сканується приймачем на побітовій основі для виявлення стартового прапорця, а потім під час прийому для виявлення стопового прапорця. Тому довжина кадра в цьому випадку не обов'язково має бути кратна 8 біт.

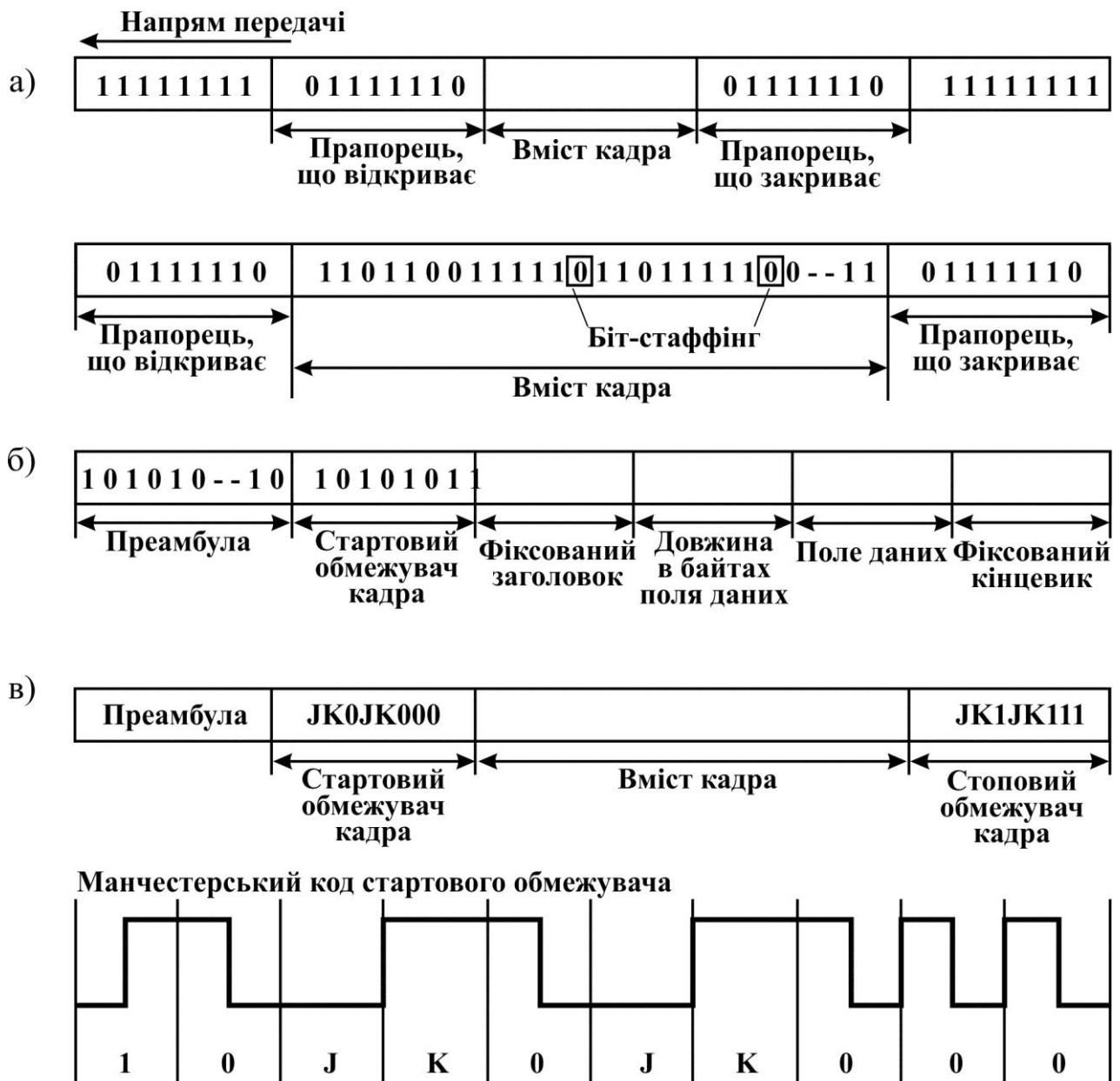


Рисунок 7.2 – Способи виділення початка і кінця кадра під час синхронної передачі

Щоб забезпечити синхронізацію приймача, передавач посилає послідовність байтів простою (кожний складається з 11111111), що передуює стартовому прапорцю.

Для досягнення прозорості даних у цій схемі необхідно, щоб прапорець не був присутній у полі даних кадра. Це досягається за допомогою прийому, відомого як вставка 0 біта, – біт-стаффі́нг. Схема вставки біта працює тільки під час передачі поля даних кадра. Якщо ця схема виявляє, що підряд передано п'ять 1, то вона автоматично вставляє додатковий 0 (навіть якщо після цих п'яти 1 ішов 0). Тому послідовність 01111110 ніколи не з'явиться

в полі даних кадра. Аналогічна схема працює в приймачі й виконує зворотну функцію. Коли після п'яти 1 виявляється 0, він автоматично видаляється з поля даних кадра.

Біт-стаффінг набагато економічніший, ніж байт-стаффінг, оскільки замість зайвого байта вставляється один біт, отже, швидкість передачі користувальницьких даних у цьому випадку вповільнюється в меншому ступені.

У другій схемі (рис. 7.2, б) для позначення початку кадра є тільки стартовий прапорець, а для визначення кінця кадра використовується поле довжини кадра, яке при фіксованих розмірах заголовка й концевика найчастіше має значення довжини поля даних кадра. Ця схема найбільш застосовувана в локальних мережах. У цих мережах для позначення факта незайнятості середовища у вихідному стані по середовищу взагалі не передається ніяких символів. Щоб всі інші станції увійшли в бітову синхронізацію, станція, яка посилає, випереджає вміст кадра послідовністю біт, відомою як преамбула, що складається із одиниць і нулів 101010... Увійшовши в бітову синхронізацію, приймач вивчає вхідний потік на побітовій основі, поки не виявить байт початку кадра 10101011, який виконує роль символу STX. За цим байтом іде заголовок кадра, у якому в певному місці знаходиться поле довжини поля даних. Отже у цій схемі приймач просто відраховує задану кількість байт, щоб визначити закінчення кадра.

Третя схема (рис. 7.2, в) використовує для позначення початку й кінця кадра прапорці, які включають заборонені для даного коду сигнали (code violations, V). Наприклад, під час манчестерського кодування замість обов'язкової зміни полярності сигналу в середині тактового інтервала рівень сигналу залишається незмінним і низьким (заборонений сигнал J) або незмінним і високим (заборонений сигнал K). Початок кадра позначається послідовністю JK0JK000, а кінець – послідовністю JK1JK100. Цей спосіб дуже економічний, оскільки не вимагає ні біт-стаффінга, ні поля довжини, але його недоліком є залежність від обраного метода фізичного кодування. Під час використання надлишкових кодів роль сигналів J і K відіграють заборонені символи, наприклад, у коді 4В/5В цими символами є коди 11000 і 10001.

Кожна з трьох схем має свої переваги й недоліки. Прапорці дозволяють відмовитися від спеціального додаткового поля, але вимагають спеціальних заходів: або з дозволу розміщення прапорця в полі даних за рахунок біт-стаффінга, або з використання як прапорця заборонених сигналів, що робить цю схему залежною від способу кодування.

Протоколи з гнучким форматом кадра

Для значної частини протоколів характерні кадри, які складаються зі службових полів фіксованої довжини. Виключення робиться тільки для поля даних, з метою ощадливого пересилання як невеликих квитанцій, так і великих файлів. Спосіб визначення закінчення кадра шляхом задання довжини поля даних, розглянутий вище, саме розрахований на такі кадри з фіксованою структурою й фіксованими розмірами службових полів.

Однак існує ряд протоколів, у яких кадри мають гнучку структуру. Наприклад, до таких протоколів належать дуже популярний прикладний протокол управління мережами SNMP, а також протокол канального рівня PPP, який використовується для з'єднань типу «точка-точка». Кадри таких протоколів складаються з невизначеної кількості полів, кожне з яких може мати змінну довжину. Початок такого кадра відзначається деяким стандартним чином, наприклад, за допомогою прапорця, а потім протокол послідовно переглядає поля кадра й визначає їхню кількість і розміри. Кожне поле зазвичай описується двома додатковими полями фіксованого розміру. Наприклад, якщо в кадрі зустрічається поле, яке містить деякий символічний рядок, то в кадр вставляються три поля:

Тип	Довжина	Значення
string	6	public

Додаткові поля «Тип» і «Довжина» мають фіксований розмір в один байт, тому протокол легко знаходить межі поля «Значення». Оскільки кількість таких полів також невідома, для визначення загальної довжини кадра використовується або загальне поле «Довжина», яке міститься на початку кадра й належить до всіх полів даних, або закриваючий прапорець.

7.3 Вибір розміру кадра

Вибір максимальної довжини кадра або максимальної довжини пакета в мережах передачі даних залежить від багатьох факторів. Один з найважливіших полягає в тому, що службова інформація включається в кожен кадр. Нехай V буде кількістю бітів службової інформації в кадрі, включаючи заголовок кадра, трейлер, прапорець і заголовок пакета, але без фіксованої службової інформації, пов'язаної з повідомленням. Нехай $K_{\max}$ – максимальна довжина пакета, не включаючи заголовок пакета, а M – довжина даного повідомлення,

включаючи фіксовану службову інформацію, пов'язану з повідомленням. Тоді загальна кількість бітів, які мали передаватися для того, щоб передати одне повідомлення, дорівнює

$$\text{Загальна кількість бітів} = M + \lceil M/K_{\max} \rceil V, \quad (7.1)$$

де $\lceil x \rceil$ позначає найменше ціле число, більше або таке, що дорівнює x .

Інакше кажучи, у міру того як $K_{\max}$ зменшується, необхідна кількість кадрів зростає й службова інформація V має повторюватися для кожного кадра. Цей фактор потім використовується як аргумент для збільшення максимальної довжини пакета. Другий момент, тісно пов'язаний із цим фактором, полягає в тому, що станція (вузол або зовнішній пункт) має виконувати певний обсяг обробки кожного кадра. Буде показано, що всі інші фактори, які впливають на вибір максимального розміру пакета, використовуються як аргументи для його зменшення.

Найважливіший з цих факторів пов'язаний з ефектом безперервної передачі по тракту (рис. 7.3). Ідея полягає в тому, що, якщо повідомлення передається як один пакет, повне повідомлення має бути прийняте на лінії (з урахуванням використання ARQ) перед початком передачі по наступній лінії. Однак, якщо повідомлення розбивається на кілька пакетів, більш ранні пакети можуть продовжувати просування своїм шляхом, коли більш пізні пакети ще передаються по першій лінії, що зменшує загальну затримку пакета.

Тепер досліджуємо загальний вплив обсягу службової інформації й ефекта безперервної передачі по тракту. Припустимо, що повідомлення довжини M розбивається в пакети максимальної довжини з останнім пакетом зазвичай меншої довжини. Припустимо, що пакети мають передаватися по j лініях однакової пропускної здатності й мережа мало завантажена, так що можна знехтувати очікуваннями у вузлах, які викликані іншими потоками. Ми нехтуємо помилками на лініях (які будуть обговорені пізніше) і не враховуємо час поширення (яке не залежить від максимальної довжини пакета). Тоді загальний час T необхідний для передачі повідомлення отримувачу, дорівнює часу доставки першого пакета по перших $j-1$ лініях плюс час передачі повного повідомлення по останній лінії (тобто якщо не останній кадр закінчує передачу по лінії, наступний кадр завжди готовий почати передачу по цій лінії). Нехай C – пропускна здатність кожної лінії в бітах за секунду, так що TC – кількість переданих бітів, необхідних для доставки повідомлення.

Тоді

$$TC = (K_{\max} + V)(j - 1) + M + \lceil M/K_{\max} \rceil V. \quad (7.2)$$

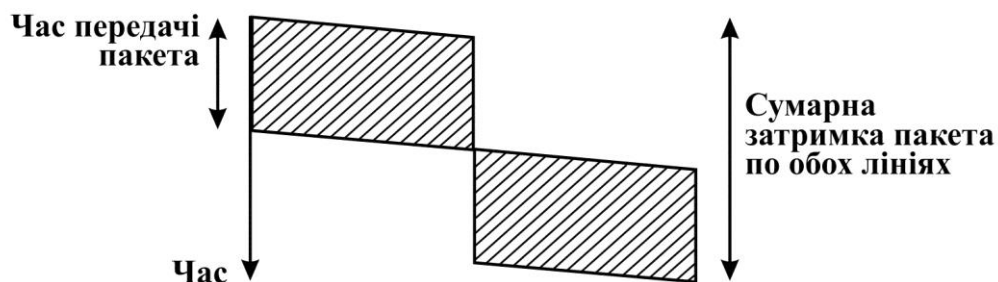
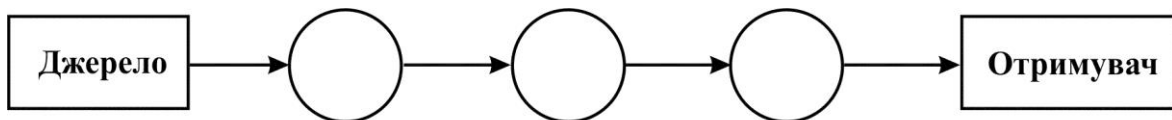
Якщо взяти математичне очікування цього виразу по довжинах повідомлень M , отримаємо, що

$$E\{\lceil M/K_{\max} \rceil\} = E\{M/K_{\max}\} + \frac{1}{2}$$

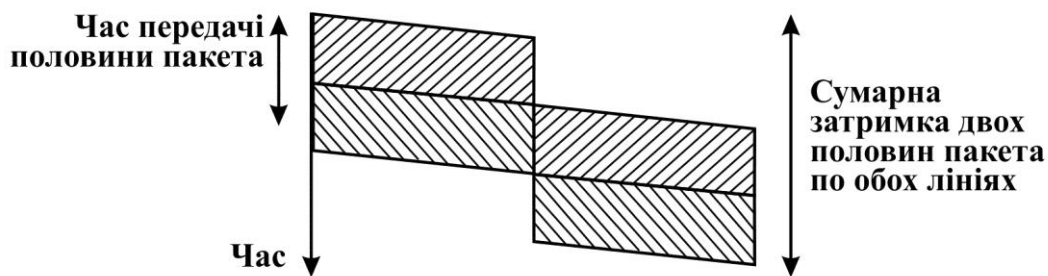
(це прийнятно, якщо розподіл M однорідний на проміжках з $K_{\max}$ біт).

Тоді

$$E(TC) \approx (K_{\max} + V)(j - 1) + E\{M\}V/K_{\max} + V/2. \quad (7.3)$$



а)



б)

Рисунок 7.3 – Зменшення затримки в результаті вкорочення пакетів

показує перевагу безперервної передачі пакетів за трактом:

а – загальна затримка пакета під час передачі по двох порожніх лініях дорівнює подвоєному часу передачі пакета по лінії плюс сумарна затримка поширення;

б – коли кожен пакет розділений на два, можна побачити ефект безперервної передачі по тракту. Сумарна затримка для двох половин пакета в 1,5 рази

більше часу передачі вихідного пакета по лінії

плюс сумарна затримка поширення

Якщо знехтувати тим, що $K_{\max}$ має бути цілим, можна продиференціювати цей вираз за $K_{\max}$ для того, щоб знайти величину $K_{\max}$, яка мінімізує $E\{TC\}$. У результаті отримуємо

$$K_{\max} \approx \sqrt{\frac{E\{M\}V}{j-1}}. \quad (7.4)$$

Це значення $K_{\max}$ є компромісним між обсягом службової інформації й ефектом безперервної передачі по тракту. У міру того як збільшується обсяг службової інформації V , величина $K_{\max}$ також збільшується, а зі збільшенням довжини шляху j величина $K_{\max}$ має зменшуватися. Тут доречно нагадати одну практичну деталь: затримка під час передачі файлів часто менш важлива, ніж під час передачі інших повідомлень, так що передача файлів має, імовірно, враховуватися з меншою вагою, ніж інші повідомлення, під час оцінки $E\{M\}$; отже, це дає трохи менше значення $K_{\max}$.

Зі збільшенням навантаження в мережі ефект безперервної передачі по тракту зберігається, незважаючи на те, що пакети зазвичай мають стояти в черзі у вузлі перед початком передачі. Однак при великих навантаженнях вплив обсягу службової інформації стає більш значним через зростання кількості бітів, які мають передаватися при малому розмірі пакета. Крім того, існують кілька інших причин при великих навантаженнях, які використовуються як аргументи для зменшення розміру пакета. Одна із цих причин – ефект «повільної вантажівки». Якщо багато пакетів з різними довжинами передаються тими самими шляхами, то короткі пакети збираються за довгими пакетами через велику затримку передачі довгих пакетів на кожній лінії; це аналогічно скупченню автомобілів за повільною вантажівкою, на однополосній дорозі. Такий самий вплив на затримку очікування в черзі для пакетів з різних шляхів проходження розглядатиметься у другій частині конспекту лекцій; велика дисперсія довжини пакетів (яка виникає при великому максимальному розмірі пакета) збільшує затримку.

У лекції 10 буде показано вплив великої дисперсії довжини кадрів на системи ARQ на n кроків назад. Велика дисперсія або збільшує кількість пакетів, які мають бути повторно передані, або збільшує незайнятий час. Це також є причиною вибору малого максимального розміру пакета. І, врешті решт, існує ефект помилок передачі. Довгі кадри мають більш високу ймовірність помилки, ніж короткі. Для більшості використовуваних ліній зв'язку (за винятком радіоліній) імовірність помилки за розумних розмірів

кадрів має порядок 10^{-4} або менше, так що цей ефект зазвичай менш важливий, ніж інші розглянуті ефекти. На жаль, у літературі описано багато методів вибору максимальної довжини кадра з урахуванням тільки цього ефекту. Ці методи можна використовувати лише в тих особливих випадках, коли ймовірність помилки велика.

На практиці максимальна довжина пакета зазвичай дорівнює декільком тисячам бітів. Якщо обсяг службової інформації можна зменшити, то розумно перейти до більш коротких максимальних довжин пакетів, послабляючи, таким чином, вплив дисперсії пакета, на яку вказувалося вище.

7.4 Контрольні запитання та завдання

1. Визначте, який кадр передасть на лінію передавач, якщо він працює з використанням техніки біт-стаффінга із прапорцем 7E, а на вхід передавача надійшла послідовність 24 A5 7E 56 8C (всі значення – шістнадцяткові)?

2. Наведіть аргументи за зменшення розміру пакетів під час передачі по мережах ліній зв'язку.

3. Опишіть вплив безперервної передачі на пропускну здатність мережі.

4. Визначте надлишковість стаффінга залежно від довжини прапорця.

Лекція 8

МЕТОДИ КОМУТАЦІЇ

8.1 Класифікація методів комутації

Будь-які мережі зв'язку підтримують деякий спосіб комутації своїх абонентів один з одним. Цими абонентами можуть бути віддалені комп'ютери, локальні мережі, факс-апарати або просто співрозмовники, що спілкуються за допомогою телефонних апаратів. Практично неможливо надати кожній парі взаємодіючих абонентів свою власну фізичну лінію зв'язку, що не комутується, якою вони могли б монополювати «володіти» протягом тривалого часу. Тому в будь-якій мережі завжди застосовується деякий спосіб комутації абонентів, який забезпечує доступність наявних фізичних каналів одночасно для декількох сеансів зв'язку між абонентами мережі. На рис. 8.1 показана типова структура мережі з комутацією абонентів.

Абоненти з'єднуються з комутаторами індивідуальними лініями зв'язку, кожна з яких використовується в будь-який момент часу тільки одним, закріпленим за цією лінією абонентом. Між комутаторами лінії зв'язку розділяються декількома абонентами, тобто використовуються спільно.

Існують три принципово різні схеми комутації абонентів у мережах: *комутація каналів (circuit switching)*, *комутація пакетів (packet switching)* і *комутація повідомлень (message switching)*. Зовні всі ці схеми відповідають наведеній на рис. 8.1 структурі мережі, однак їхні можливості й властивості різні.

Мережі з комутацією каналів мають багату історію, бо походять від перших телефонних мереж. Мережі з комутацією пакетів з'явилися наприкінці 60-х років як результат експериментів з першими глобальними комп'ютерними мережами. Мережі з комутацією повідомлень послужили прототипом сучасних мереж з комутацією пакетів, і сьогодні вони в чистому вигляді практично не існують. Кожна із цих схем має свої переваги й недоліки, але за прогнозами багатьох фахівців майбутнє належить технології комутації пакетів, як більш гнучкій та універсальній.

Як мережі з комутацією пакетів, так і мережі з комутацією каналів можна поділити на два класи за іншою ознакою – на мережі з *динамічною комутацією* й мережі з *постійною комутацією*.

У першому випадку мережа дозволяє встановлювати з'єднання з ініціативи користувача мережі. Комутація виконується на час сеансу зв'язку,

а потім (знову ж з ініціативи одного з взаємодіючих користувачів) зв'язок розривається. У загальному випадку будь-який користувач мережі може з'єднатися з будь-яким іншим користувачем мережі. Зазвичай період з'єднання між парою користувачів при динамічній комутації становить від декількох секунд до декількох годин і завершується при виконанні певної роботи – передачі файла, перегляду сторінки тексту або зображення тощо.

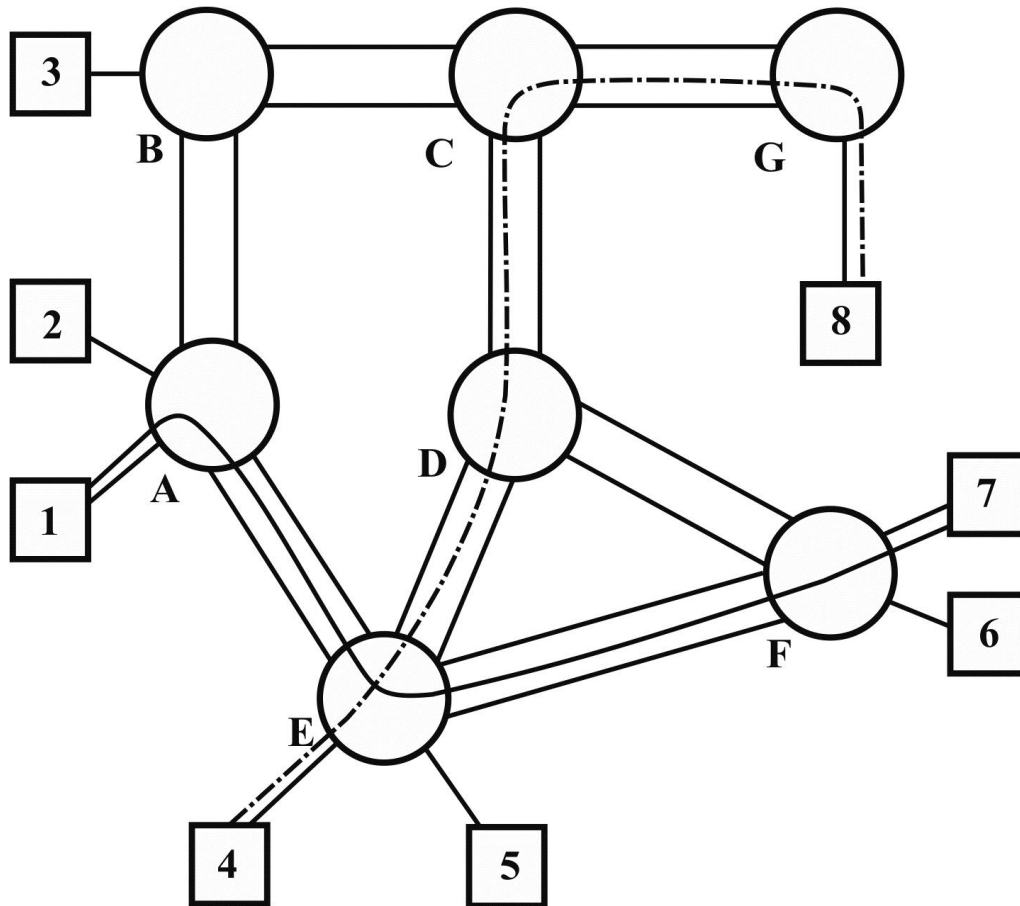


Рисунок 8.1 – Загальна структура мережі з комутацією абонентів

У другому випадку мережа не надає користувачу можливість виконати динамічну комутацію з іншим довільним користувачем мережі. Замість цього мережа дозволяє парі користувачів замовити з'єднання на тривалий період часу. З'єднання встановлюється не користувачами, а персоналом, що обслуговує мережу. Час, на який встановлюється постійна комутація, вимірюється зазвичай декількома місяцями. Режим постійної комутації в мережах з комутацією каналів часто називається сервісом *виділених (dedicated)* або *орендованих (leased)* каналів.

Прикладами мереж, які підтримують режим динамічної комутації, є телефонні мережі загального користування, локальні мережі, мережі TCP/IP.

Найпопулярнішими мережами, що працюють у режимі постійної комутації, сьогодні є мережі технології SDH, на основі яких будуються виділені канали зв'язку з пропускнуою здатністю в декілька гігабіт у секунду.

Деякі типи мереж підтримують обидва режими роботи. Наприклад, мережі X.25 і АТМ можуть надавати користувачу можливість динамічно зв'язатися з будь-яким іншим користувачем мережі, й у той самий час відправляти дані по постійному з'єднанню одному певному абоненту.

8.2 Комутація каналів

Комутація каналів має на увазі утворення безперервного складеного фізичного каналу з послідовно з'єднаних окремих каналних ділянок для прямої передачі даних між вузлами. Окремі канали з'єднуються між собою спеціальною апаратурою – комутаторами, які можуть встановлювати зв'язки між будь-якими кінцевими вузлами мережі. У мережі з комутацією каналів перед передачею даних завжди необхідно виконати процедуру встановлення з'єднання, у процесі якої й створюється складений канал.

Наприклад, якщо мережа, зображена на рис. 8.1, працює за технологією комутації каналів, то вузол 1, щоб передати дані вузлу 7, насамперед має передати спеціальний запит на встановлення з'єднання комутатору А, вказавши адресу призначення 7. Комутатор А має обрати маршрут утворення складеного каналу, а потім передати запит наступному комутатору, у цьому випадку Е. Потім комутатор Е передає запит комутатору F, а той, у свою чергу, передає запит вузлу 7. Якщо вузол 7 приймає запит на встановлення з'єднання, він направляє по вже встановленому каналу відповідь вихідному вузлу, після чого складений канал вважається зкомутованим і вузли 1 і 7 можуть обмінюватися по ньому даними, наприклад, вести телефонну розмову.

Комутатори, а також канали, що їхні з'єднують, мають забезпечувати одночасну передачу даних декількох абонентських каналів. Для цього вони мають бути високошвидкісними й підтримувати яку-небудь техніку мультиплексування абонентських каналів.

На сьогодні для мультиплексування абонентських каналів використовується дві техніки:

- техніка частотного мультиплексування (Frequency Division Multiplexing, FDM);
- техніка мультиплексування з поділом часу (Time Division Multiplexing, TDM).

Комутація каналів на основі частотного мультиплексування

Техніка частотного мультиплексування каналів (FDM) була розроблена для телефонних мереж, але застосовується вона й для інших видів мереж, наприклад мереж кабельного телебачення.

Розглянемо особливості цього виду мультиплексування на прикладі телефонної мережі.

Мовні сигнали мають спектр шириною приблизно в 10000 Гц, однак основні гармоніки укладаються в діапазон від 300 до 3400 Гц. Тому для якісної передачі мови досить утворити між двома співрозмовниками канал зі смугою пропускання в 3100 Гц, що й використовується в телефонних мережах для з'єднання двох абонентів. У той самий час смуга пропускання кабельних систем з проміжними підсилювачами, що з'єднують телефонні комутатори між собою, зазвичай становить сотні кілогерців, а іноді й сотні мегагерц. Однак безпосередньо передавати сигнали декількох абонентських каналів по широкосмуговому каналу неможливо, оскільки всі вони працюють в одному й тому самому діапазоні частот і сигнали різних абонентів змішаються між собою так, що розділити їх буде неможливо.

Для поділу абонентських каналів характерна техніка модуляції високочастотного несучого синусоїдального сигналу низькочастотним мовним сигналом (рис. 8.2). Ця техніка подібна до техніки аналогової модуляції під час передачі дискретних сигналів модемами, тільки замість дискретного вихідного сигналу використовуються безперервні сигнали, які породжуються звуковими коливаннями. У результаті спектр модульованого сигналу переноситься в інший діапазон, який симетрично розташовується щодо несучої частоти й має ширину, яка приблизно збігається з шириною модулюючого сигналу.

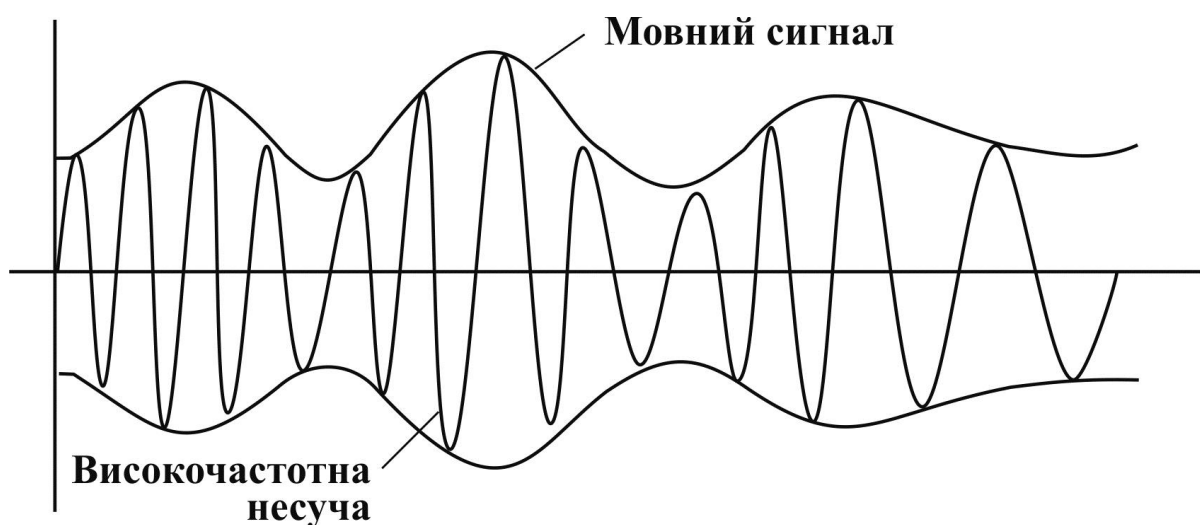


Рисунок 8.2 – Модуляція мовним сигналом

Якщо сигнали кожного абонентського каналу перенести у свій власний діапазон частот, то в одному широкосмуговому каналі можна одночасно передавати сигнали декількох абонентських каналів.

На входи FDM-комутатора надходять вихідні сигнали від абонентів телефонної мережі. Комутатор виконує перенесення частоти кожного каналу у свій діапазон частот. Зазвичай високочастотний діапазон поділяється на смуги, які приділяються для передачі даних абонентських каналів (рис. 8.3). Щоб низькочастотні складові сигналів різних каналів не змішувалися між собою, смуги роблять шириною в 4 кГц, а не в 3,1 кГц, залишаючи між ними страховий проміжок в 900 Гц. У каналі між двома FDM-комутаторами водночас передаються сигнали всіх абонентських каналів, але кожен з них займає свою смугу частот. Такий канал називають *ущільненим*.

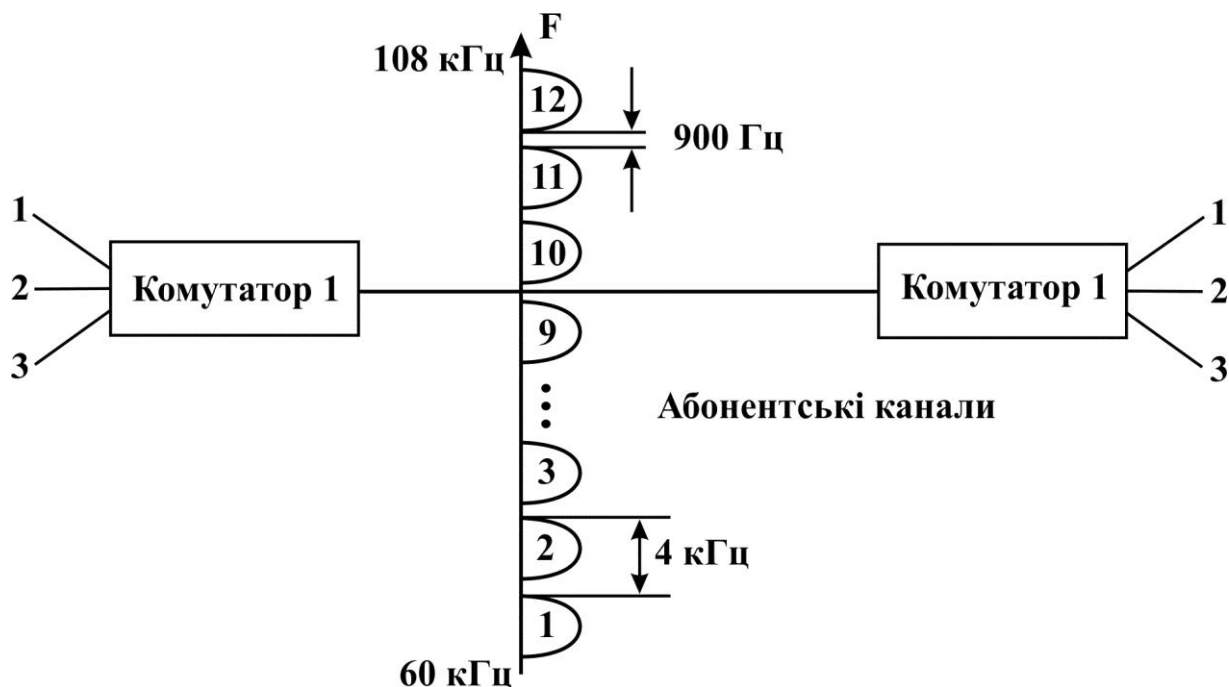


Рисунок 8.3 – Комутація на основі частотного ущільнення

Вихідний FDM-комутатор виділяє модульовані сигнали кожної несучої частоти й передає їх на відповідний вихідний канал, до якого безпосередньо підключений абонентський телефон.

У мережах на основі FDM-комутації прийнято кілька рівнів ієрархії ущільнених каналів. Перший рівень ущільнення утворюють 12 абонентських каналів, які становлять *базову групу* каналів, що займає смугу частот шириною в 48 кГц з межами від 60 до 108 кГц. Другий рівень ущільнення утворюють 5 базових груп, які становлять *супергрупу*, зі смугою частот шириною в 240 кГц і межами від 312 до 552 кГц. Супергрупа передає дані 60 абонентських каналів

тональної частоти. Десять супергруп утворюють *головну групу*, яка використовується для зв'язку між комутаторами на великих відстанях. Головна група передає дані 600 абонентів водночас і вимагає від каналу зв'язку смугу пропускання шириною не менше 2520 кГц з межами від 564 до 3084 кГц.

Комутатори FDM можуть виконувати як динамічну, так і постійну комутацію. При динамічній комутації один абонент ініціює з'єднання з іншим абонентом, посилаючи в мережу номер абонента, який викликається. Комутатор динамічно виділяє даному абоненту одну з вільних смуг свого ущільненого каналу. При постійній комутації за абонентом смуга в 4 кГц закріплюється на тривалий термін шляхом налаштування комутатора по окремому входу, недоступному користувачам.

Принцип комутації на основі поділу частот залишається незмінним і в мережах іншого вигляду, змінюються тільки межі смуг, які виділяються окремому абонентському каналу, а також кількість низькошвидкісних каналів в ущільненому високошвидкісному.

Комутація каналів на основі поділу часу

Комутація на основі техніки поділу частот розроблялася в розрахунку на передачу безперервних сигналів, які представляють голос. В ході переходу до цифрової форми подання голосу була розроблена нова техніка мультиплексування, яка орієнтується на дискретний характер переданих даних.

Ця техніка зветься *мультиплексування з поділом часу (Time Division Multiplexing, TDM)*. Рідше використовується й інша її назва – *техніка синхронного режиму передачі (Synchronous Transfer Mode, STM)*. Рисунок 8.4 пояснює принцип комутації каналів на основі техніки TDM.

Апаратура TDM-мереж – мультиплексори, комутатори, демультимплексори – працює в режимі поділу часу, по черзі обслуговуючи протягом циклу своєї роботи всі абонентські канали. Цикл роботи обладнання TDM дорівнює 125 мкс, що відповідає періоду проходження вимірів голосу в цифровому абонентському каналі. Це означає, що мультиплексор або комутатор встигає вчасно обслужити будь-який абонентський канал і передати його чергове вимірювання далі по мережі. Кожному з'єднанню виділяється один квант часу циклу роботи апаратури, який називається також тайм-слотом. Тривалість тайм-слота залежить від кількості абонентських каналів, які обслуговуються мультиплексором TDM або комутатором.

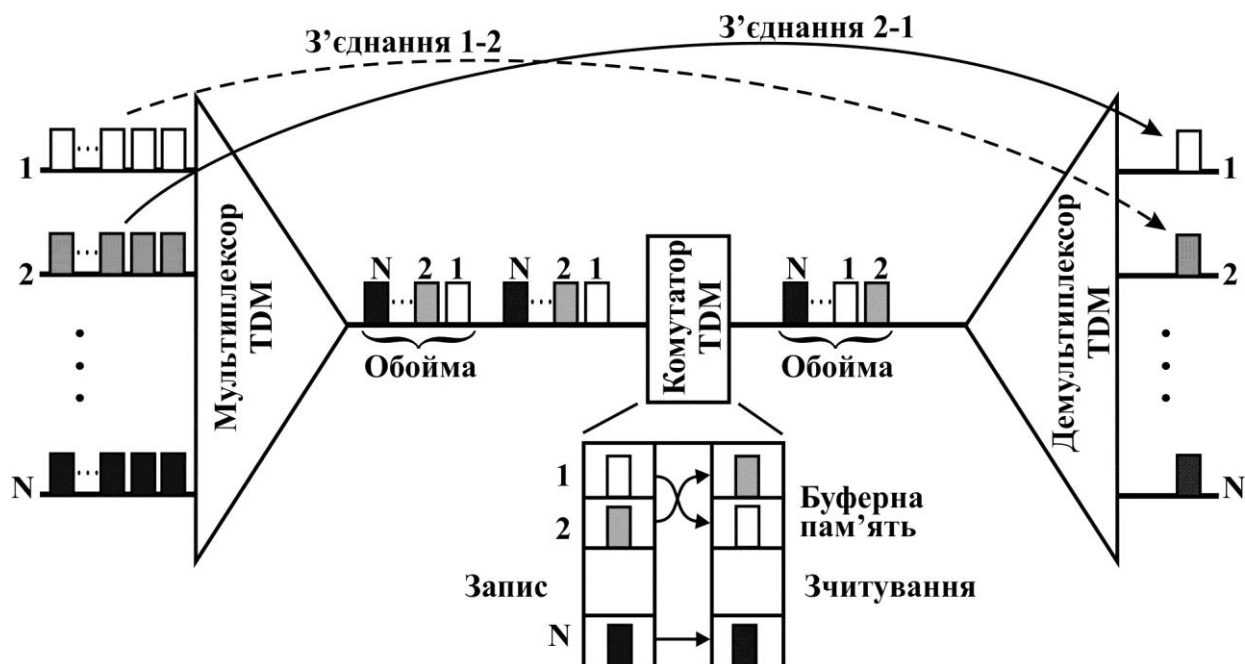


Рисунок 8.4 – Комутація на основі поділу каналу в часі

Мультиплексор приймає інформацію з N вхідних каналів від кінцевих абонентів, кожен з яких передає дані абонентському каналу зі швидкістю 64 Кбіт/с – 1 байт кожні 125 мкс. У кожному циклі мультиплексор виконує такі дії:

- прийом від кожного каналу чергового байта даних;
- складання із прийнятих байтів ущільненого кадра, який називається також обоймою;
- передача ущільненого кадра на вихідний канал з бітовою швидкістю, яка дорівнює $N \times 64$ Кбіт/с.

Порядок байт в обоймі відповідає номеру вхідного каналу, від якого цей байт отримано. Кількість абонентських каналів, які обслуговуються мультиплексором, залежить від його швидкодії. Наприклад, мультиплексор T1, який є першим промисловим мультиплексором, що працював за технологією TDM, підтримує 24 вхідних абонентських канали, створюючи на виході обойми стандарту T1, передані з бітовою швидкістю 1,544 Мбіт/с.

Демультимплексор виконує зворотну задачу – він розбирає байти ущільненого кадра і розподіляє їх за своїми декількома вихідними каналами, при цьому він вважає, що порядковий номер байта в обоймі відповідає номеру вхідного каналу.

Комутатор приймає ущільнений кадр по швидкісному каналу від мультиплексора й записує кожний байт із нього в окрему комірку своєї буферної пам'яті, причому в тому порядку, у якому ці байти були упаковані

в ущільнений кадр. Для виконання операції комутації байти витягують із буферної пам'яті не в порядку надходження, а в такому порядку, який відповідає з'єднанням абонентів, що підтримуються у мережі. Так, наприклад, якщо перший абонент лівої частини мережі рис. 8.4 має з'єднатися з другим абонентом у правій частині мережі, то байт, записаний у перший осередок буферної пам'яті, витягатиме з її другим. «Перемішуючи» потрібним чином байти в обоймі, комутатор забезпечує з'єднання кінцевих абонентів у мережі.

Одного разу виділений номер тайм-слота залишається у розпорядженні з'єднання «вхідний канал – вихідний слот» протягом усього часу існування цього з'єднання, навіть якщо переданий трафік є пульсуючим і не завжди потребує захопленої кількості тайм-слотів. Це означає, що з'єднання в мережі TDM завжди володіє відомою й фіксованою пропускну здатністю, кратною 64 Кбіт/с.

Робота обладнання TDM нагадує роботу мереж з комутацією пакетів, оскільки кожен байт даних можна вважати деяким елементарним пакетом. Однак, на відміну від пакета комп'ютерної мережі, «пакет» мережі TDM не має індивідуальної адреси. Його адресою є порядковий номер в обоймі або номер виділеного тайм-слота у мультиплексорі або комутаторі. Мережі, що використовують техніку TDM, вимагають синхронної роботи всього обладнання, що й визначило другу назву цієї техніки – синхронний режим передачі (STM). Порушення синхронності руйнує необхідну комутацію абонентів, оскільки при цьому губиться адресна інформація. Тому перерозподіл тайм-слотів між різними каналами в обладнанні TDM неможливий, навіть якщо в якомусь циклі роботи мультиплексора тайм-слот одного з каналів виявляється надлишковим, оскільки на вході цього каналу в цей момент немає даних для передачі (наприклад, абонент телефонної мережі мовчить).

Існує модифікація техніки TDM, яка називається *статистичним поділом каналу в часі (Statistical TDM, STDM)*. Ця техніка розроблена спеціально для того, щоб за допомогою тимчасово вільних тайм-слотів одного каналу можна було збільшити пропускну здатність інших. Для вирішення цієї задачі кожен байт даних доповнюється полем адреси невеликої довжини, наприклад, в 4 або 5 біт, що дозволяє мультиплексувати 16 або 32 канали. Однак техніка STDM не знайшла широкого застосування й використовується в основному в нестандартному обладнанні підключення терміналів до мейнфреймів. Розвитком ідей статистичного мультиплексування стала технологія асинхронного режиму передачі – АТМ, що увібрала в себе кращі риси техніки комутації каналів і пакетів.

Мережі TDM можуть підтримувати або режим динамічної комутації, або режим постійної комутації, а іноді й обидва режими. Так, наприклад, основним режимом цифрових телефонних мереж, що працюють на основі технології TDM, є динамічна комутація, але вони підтримують також і постійну комутацію, надаючи своїм абонентам службу виділених каналів.

Існує апаратура, яка підтримує тільки режим постійної комутації. До неї належить обладнання типу T1/E1, а також високошвидкісне обладнання SDH. Таке обладнання використовується для побудови первинних мереж, основною функцією яких є створення виділених каналів між комутаторами, які підтримують динамічну комутацію.

Сьогодні практично всі дані – голос, зображення, комп'ютерні дані – передаються в цифровій формі. Тому виділені канали TDM-технології, які забезпечують нижній рівень для передачі цифрових даних, є універсальними каналами для побудови мереж будь-якого типу: телефонних, телевізійних і комп'ютерних.

Загальні властивості мереж з комутацією каналів

Мережі з комутацією каналів мають декілька важливих загальних властивостей незалежно від того, який тип мультиплексування в них використовується.

Мережі з динамічною комутацією вимагають попередньої процедури встановлення з'єднання між абонентами. Для цього в мережу передається адреса викликуваного абонента, що проходить через комутатори й набуває їх на наступну передачу даних. Запит на встановлення з'єднання маршрутизується від одного комутатора до іншого й зрештою досягає викликуваного абонента. Мережа може відмовити у встановленні з'єднання, якщо ємність необхідного вихідного каналу вже вичерпана. Для FDM-комутатора ємність вихідного каналу дорівнює кількості частотних смуг цього каналу, а для TDM-комутатора – кількості тайм-слотів, на які ділиться цикл роботи каналу. Мережа відмовляє в з'єднанні також у тому випадку, якщо запитуваний абонент уже встановив з'єднання з будь-ким іншим. У першому випадку говорять, що зайнято комутатор, а в другому – абонент. Можливість відмови у з'єднанні є недоліком методу комутації каналів.

Якщо з'єднання може бути встановлено, то йому виділяється фіксована смуга частот в FDM-мережах або ж фіксована пропускна здатність в TDM-мережах. Ці величини залишаються незмінними протягом усього періоду з'єднання. Гарантована пропускна здатність мережі після встановлення з'єднання є

важливою властивістю, необхідною для таких додатків, як передача голосу, зображення або управління об'єктами в реальному масштабі часу. Однак динамічно змінювати пропускну здатність каналів на вимогу абонента мережі з комутацією каналів не можуть, що робить їх неефективними в умовах пульсуючого трафіка.

Недоліком мереж з комутацією каналів є неможливість застосування користувальницької апаратури, яка працює з різною швидкістю. Окремі частини складеного каналу працюють з однаковою швидкістю, оскільки мережі з комутацією каналів не буферизують даних користувачів.

Мережі з комутацією каналів добре пристосовані для комутації потоків даних постійної швидкості, коли одиницею комутації є не окремий байт або пакет даних, а довгостроковий синхронний потік даних між двома абонентами. Для таких потоків мережі з комутацією каналів додають мінімум службової інформації для маршрутизації даних через мережу, використовуючи часову позицію кожного біта потоку як його адреси призначення в комутаторах мережі.

Забезпечення дуплексного режиму роботи на основі технологій FDM, TDM і WDM

Залежно від напрямку можливої передачі дані способи передачі даних по лінії зв'язку поділяються на такі типи:

- *симплексний* – передача здійснюється по лінії зв'язку тільки в одному напрямі;
- *напівдуплексний* – передача ведеться в обох напрямках, але поперемінно в часі. Прикладом такої передачі слугує технологія Ethernet;
- *дуплексний* – передача ведеться одночасно у двох напрямках.

Дуплексний режим – найуніверсальніший і найпродуктивніший спосіб роботи каналу. Найпростішим варіантом організації дуплексного режиму є використання двох незалежних фізичних каналів (двох пар провідників або двох світловодів) у кабелі, кожний з яких працює в симплексному режимі, тобто передає дані в одному напрямі. Саме така ідея лежить в основі реалізації дуплексного режиму роботи в багатьох мережних технологіях, наприклад, Fast Ethernet або ATM.

Іноді таке просте рішення виявляється недоступним або неефективним. Найчастіше це відбувається в тих випадках, коли для дуплексного обміну даними є всього один фізичний канал, а організація другого пов'язана з великими витратами. Наприклад, під час обміну даними за допомогою модемів через телефонну мережу у користувача є тільки один фізичний канал зв'язку

з АТС – двопроводова лінія, і здобувати другий навряд чи доцільно. У таких випадках дуплексний режим роботи організовується на основі поділу каналу на два логічних підканала за допомогою техніки FDM або TDM.

Модеми для організації дуплексного режиму роботи на двопроводовій лінії застосовують техніку FDM. Модеми, які використовують частотну модуляцію, працюють на чотирьох частотах: дві частоти – для кодування одиниць і нулів в одному напрямі, а інші дві – для передачі даних у зворотному.

Під час цифрового кодування дуплексний режим на двопроводовій лінії організовується за допомогою техніки TDM. Частина тайм-слотів використовується для передачі даних в одному напрямі, а частина – в іншому. Зазвичай тайм-слоти протилежних напрямів чергуються, через що такий спосіб іноді називають «пінг-понговою» передачею. TDM-поділ лінії характерний, наприклад, для цифрових мереж з інтеграцією послуг (ISDN) на абонентських двопроводових закінченнях.

У волоконно-оптичних кабелях при використанні одного оптичного волокна для організації дуплексного режиму роботи застосовується передача даних в одному напрямі за допомогою світлового пучка однієї довжини хвилі, а у зворотному – іншої довжини хвилі. Така техніка належить до методу FDM, але для оптичних кабелів вона отримала назву *поділу по довжині хвилі (Wave Division Multiplexing, WDM)*. WDM застосовується й для підвищення швидкості передачі даних в одному напрямі, зазвичай використовуючи від 2 до 16 каналів.

8.3 Комутація пакетів

Комутація пакетів – це техніка комутації абонентів, що була спеціально розроблена для ефективної передачі комп'ютерного трафіка. Експерименти зі створення перших комп'ютерних мереж на основі техніки комутації каналів показали, що цей вид комутації не дозволяє досягти високої загальної пропускної здатності мережі. Сутність проблеми полягає в пульсуючому характері трафіка, який генерують типові мережні додатки. Наприклад, під час звернення до віддаленого файлового сервера користувач спочатку переглядає вміст каталогу цього сервера, який породжує передачу невеликого обсягу даних. Потім він відкриває необхідний файл у текстовому редакторі, і ця операція може створити досить інтенсивний обмін даними, особливо якщо файл містить об'ємні графічні домішки. Після відображення декількох сторінок файла користувач деякий час працює з ними локально, що взагалі не вимагає

передачі даних мережею, а потім повертає модифіковані копії сторінок на сервер, – і це знову породжує інтенсивну передачу даних по мережі.

Коефіцієнт пульсації трафіка окремого користувача мережі, дорівнює відношенню середньої інтенсивності обміну даними до максимально можливого, може становити 1:50 або 1:100. Якщо для описаної сесії організувати комутацію каналу між комп'ютером користувача й сервером, то більшу частину часу канал простоюватиме. Водночас комутаційні можливості мережі використовуватимуться – частина тайм-слотів або частотних смуг комутаторів буде зайнята й недоступна іншим користувачам мережі.

Під час комутації пакетів всі передані користувачем мережі повідомлення розбиваються у вихідному вузлі на порівняно невеликі частини, які називаються пакетами. Нагадаємо, що повідомленням називається логічно завершена порція даних – запит на передачу файлу, відповідь на цей запит, яка містить весь файл тощо. Повідомлення можуть мати довільну довжину, від декількох байт до багатьох мегабайт. Навпаки, пакети зазвичай теж можуть мати змінну довжину, але у вузьких межах, наприклад, від 46 до 1500 байт. Кожен пакет забезпечується заголовком, у якому вказується адресна інформація, необхідна для доставки пакета вузлу призначення, а також номер пакета, який використовуватиметься вузлом призначення для складання повідомлення (рис. 8.5). Пакети транспортуються в мережі як незалежні інформаційні блоки. Комутатори мережі приймають пакети від кінцевих вузлів і на підставі адресної інформації передають їх один одному, а в остаточному підсумку – вузлу призначення.

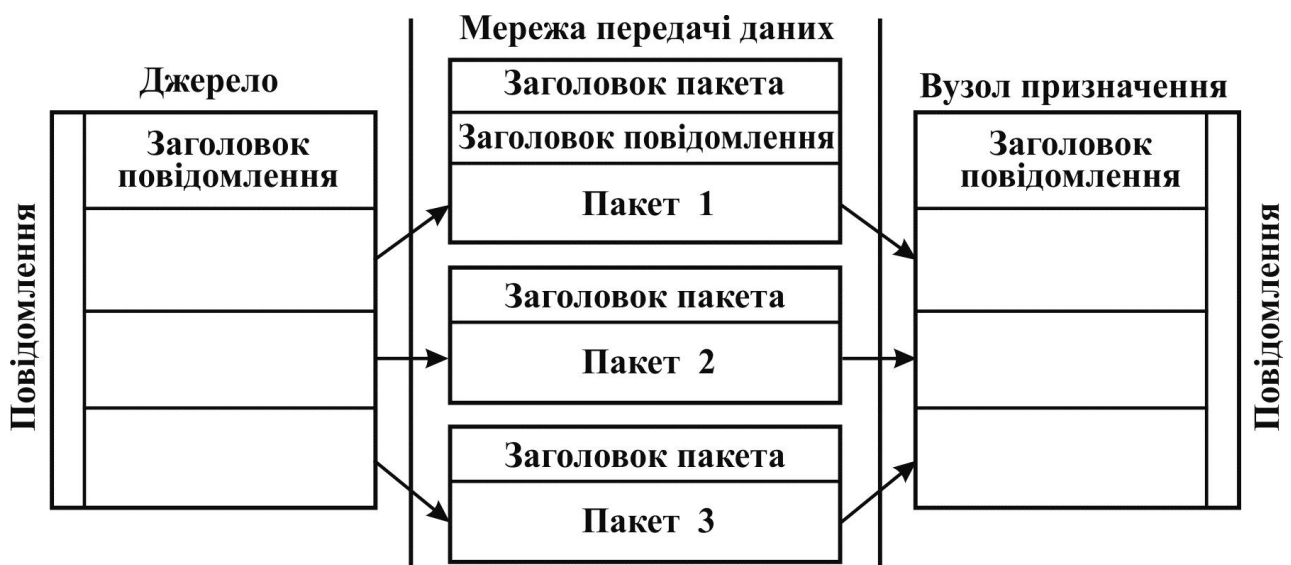


Рисунок 8.5 – Розбивка повідомлення на пакети

Комутатори пакетної мережі відрізняються від комутаторів каналів тим, що вони мають внутрішню буферну пам'ять для тимчасового зберігання пакетів, якщо вихідний порт комутатора в момент прийняття пакета зайнятий передачею іншого пакета (рис. 8.6). У цьому випадку пакет перебуває якийсь час у черзі пакетів у буферній пам'яті вихідного порту, а коли до нього дійде черга, то він передається наступному комутатору. Така схема передачі даних дозволяє згладжувати пульсації трафіка на магістральних зв'язках між комутаторами й тим самим використовувати їх найефективніше для підвищення пропускної здатності мережі в цілому.

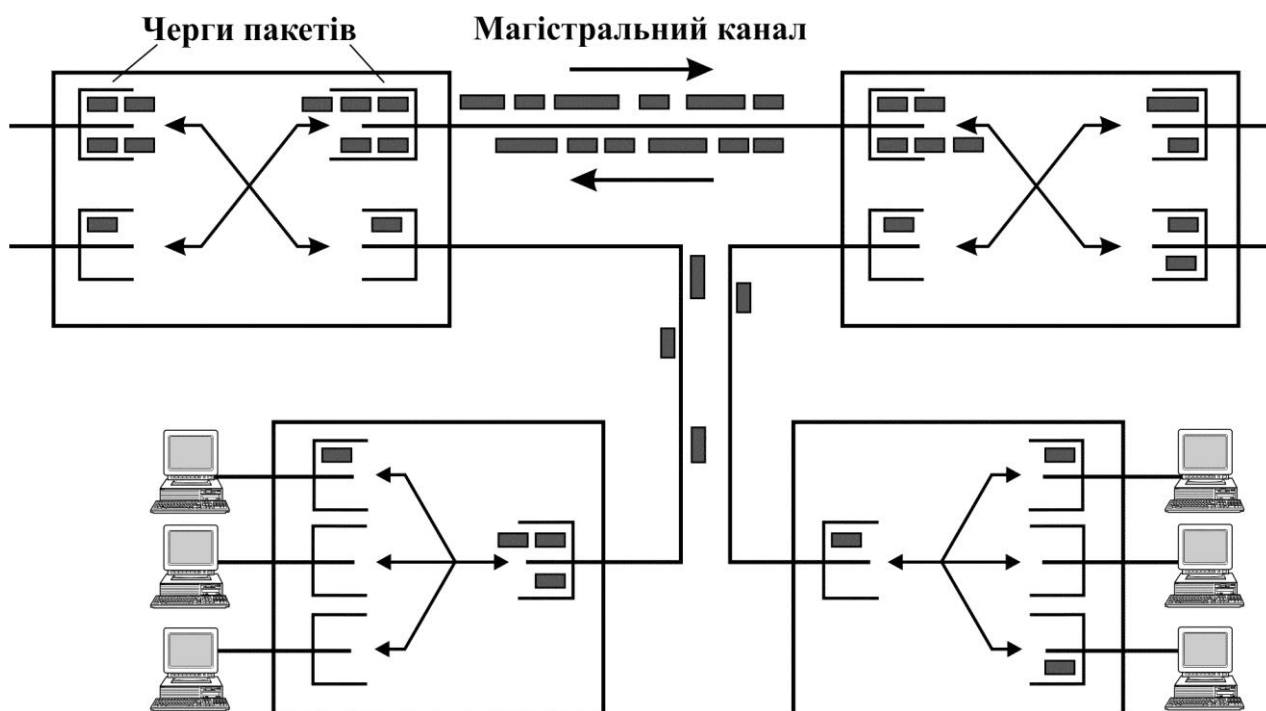


Рисунок 8.6 – Згладжування пульсацій трафіка в мережі з комутацією пакетів

Дійсно, для пари абонентів найефективнішим було б надання їм в одноособове користування скомутованого каналу зв'язку, як це робиться в мережах з комутацією каналів. При цьому способі час взаємодії цієї пари абонентів був би мінімальним, оскільки дані без затримок передавалися б від одного абонента другому. Простої каналу під час пауз передачі абонентів не цікавлять, для них важливо швидше вирішити свою власну задачу. Мережа з комутацією пакетів сповільнює процес взаємодії конкретної пари абонентів, оскільки їхні пакети можуть очікувати в комутаторах, поки по магістральних зв'язках передаються інші пакети, що прийшли в комутатор раніше.

Проте загальний обсяг переданих мережею комп'ютерних даних в одиницю часу при техніці комутації пакетів буде вищим, ніж при техніці комутації каналів, оскільки пульсації окремих абонентів відповідно до закону великих чисел розподіляються в часі. Тому комутатори постійно й досить рівномірно завантажені роботою, якщо кількість абонентів, що обслуговуються ними, дійсно багато. На рис. 8.6 показано, що трафік, який надходить від кінцевих вузлів на комутатори, дуже нерівномірно розподілений у часі. Однак комутатори вищого рівня ієрархії, які обслуговують з'єднання між комутаторами нижнього рівня, завантажені більш рівномірно, і потік пакетів у магістральних каналах, що з'єднує комутатори верхнього рівня, має майже максимальний коефіцієнт використання.

Більш висока ефективність мереж з комутацією пакетів порівняно з мережами з комутацією каналів (за рівної пропускної здатності каналів зв'язку) була доведена як експериментально, так і за допомогою імітаційного моделювання. Тут доречна аналогія з мультипрограмними операційними системами. Кожна окрема програма в такій системі виконується довше, ніж в однопрограмній, коли програмі виділяється весь процесорний час, поки вона не завершить своє виконання. Однак загальна кількість програм, що виконуються за одиницю часу, у мультипрограмній системі більша, ніж в однопрограмній.

8.4 Контрольні запитання та завдання

1. Поясніть, у чому проявляється надмірність TDM-технології?
2. Назвіть, який спосіб комутації ефективніший: комутація каналів або комутація пакетів?
3. Поясніть, у чому відмінність комутаторів пакетної мережі від комутаторів каналів?
4. Поясніть, чому техніка комутації каналів не дозволяє досягти високої загальної пропускної здатності мережі?
5. Поясніть, чому в мережах з комутацією каналів неможливе застосування користувальницької апаратури, що працює з різною швидкістю?

Лекція 9

МЕТОДИ КОМУТАЦІЇ ПАКЕТІВ

9.1 Комутація пакетів з використанням методу віртуальної передачі

Описаний вище режим передачі пакетів між двома кінцевими вузлами мережі припускає незалежну маршрутизацію кожного пакета. Такий режим роботи мережі називається дейтаграмним, і в ході його використання комутатор може змінити маршрут будь-якого пакета залежно від стану мережі – працездатності каналів і інших комутаторів, довжини черг пакетів у сусідніх комутаторах тощо.

Існує й інший режим роботи мережі – передача пакетів по *віртуальному каналу* (*virtual circuit* або *virtual channel*). У цьому випадку перед тим, як почати передачу даних між двома кінцевими вузлами, має бути встановлений віртуальний канал, який є єдиним маршрутом, що з'єднує ці кінцеві вузли. Віртуальний канал може бути динамічним або постійним. Динамічний віртуальний канал встановлюється під час передачі в мережу спеціального пакета – запита на встановлення з'єднання. Цей пакет проходить через комутатори й «прокладає» віртуальний канал. Це означає, що комутатори запам'ятовують маршрут для даного з'єднання й при надходженні наступних пакетів даного з'єднання відправляють їх завжди прокладеним маршрутом. Постійні віртуальні канали створюються адміністраторами мережі шляхом ручного налаштування комутаторів.

При відмові комутатора або каналу на шляху віртуального каналу з'єднання розривається, і віртуальний канал потрібно прокладати заново. При цьому він, природно, обійде ділянки мережі, які відмовили.

Кожен режим передачі пакетів має свої переваги й недоліки. Дейтаграмний метод не вимагає попереднього встановлення з'єднання й тому працює без затримки перед передачею даних. Це особливо вигідно для передачі невеликого обсягу даних, коли час встановлення з'єднання може бути порівнянним з часом передачі даних. Крім того, дейтаграмний метод швидше адаптується до змін у мережі.

В ході використання методу віртуальних каналів час, витрачений на встановлення віртуального каналу, компенсується наступною швидкою передачею всього потоку пакетів. Комутатори розпізнають належність пакета до віртуального каналу за спеціальною позначкою – номером віртуального

каналу, а не аналізують адреси кінцевих вузлів, як це робиться при дейтаграмному методі.

Пропускна здатність мереж з комутацією пакетів

Однією з відмінностей методу комутації пакетів від методу комутації каналів є невизначеність пропускної здатності з'єднання між двома абонентами. У методі комутації каналів після утворення складеного каналу пропускна здатність мережі при передачі даних між кінцевими вузлами відома – це пропускна здатність каналу. Дані після затримки, пов'язаної з установленням каналу, починають передаватися на максимальній для каналу швидкості (рис. 9.1, а). Час передачі повідомлення в мережі з комутацією каналів $T_{ц.к.}$ дорівнює сумі затримки поширення сигналу по лінії зв'язку $t_{a.p.}$ і затримки передачі повідомлення $t_{з.н.}$. Затримка поширення сигналу залежить від швидкості поширення електромагнітних хвиль у конкретному фізичному середовищі, що коливається від 0,6 до 0,9 швидкості світла у вакуумі. Час передачі повідомлення дорівнює V/C , де V – обсяг повідомлення в бітах, а C – пропускна здатність каналу в бітах у секунду.

У мережі з комутацією пакетів спостерігається принципово інша картина.

Процедура встановлення з'єднання в цих мережах, якщо вона використовується, займає приблизно такий же час, як і в мережах з комутацією каналів, тому порівнюватимемо тільки час передачі даних.

На рис. 9.1, б показано приклад передачі в мережі з комутацією пакетів. Припускається, що в мережу передається повідомлення того ж обсягу, що й повідомлення, яке ілюструє рис. 9.1, а, однак воно розділене на пакети, кожен з яких забезпечений заголовком. Час передачі повідомлення в мережі з комутацією пакетів позначено на рисунку $T_{к.п.}$. В ході передачі цього повідомлення, розбитого на пакети, по мережі з комутацією пакетів виникають додаткові часові затримки.

По-перше, це затримки в джерелі передачі, яке, крім передачі власне повідомлення, витрачає додатковий час на передачу заголовків $t_{н.з.}$, плюс до цього додаються затримки $t_{инт}$, викликані інтервалами між передачею кожного наступного пакета (цей час іде на формування чергового пакета стеком протоколів).

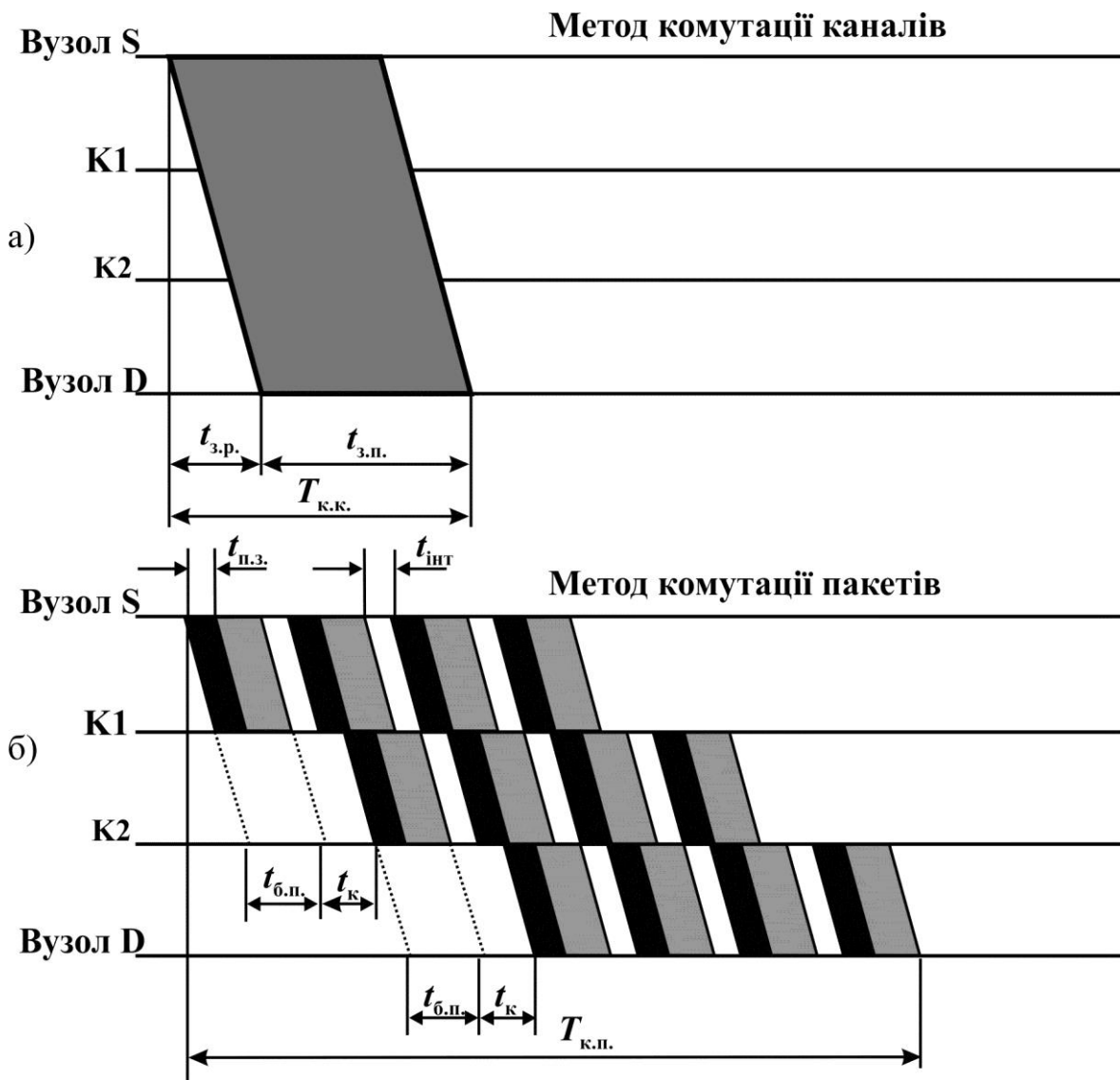


Рисунок 9.1 – Затримки передачі даних у мережах з комутацією каналів і пакетів

По-друге, додатковий час витрачається в кожному комутаторі. Тут затримки складаються з часу буферизації пакета $t_{б.п.}$ (комутатор не може почати передачу пакета, не прийнявши його повністю у свій буфер) і часу комутації $t_{к.}$. Час буферизації дорівнює часу прийому пакета з бітовою швидкістю протокола. Час комутації складається з часу очікування пакета в черзі й часу переміщення пакета у вихідний порт. Якщо час переміщення пакета фіксований і зазвичай невеликий (від декількох мікросекунд до декількох десятків мікросекунд), то час очікування пакета в черзі коливається в дуже широких межах і заздалегідь невідомий, оскільки залежить від поточного завантаження мережі пакетами.

Проведемо грубу оцінку затримки в передачі даних у мережах з комутацією пакетів порівняно з мережами з комутацією каналів на найпростішому

прикладі. Нехай тестове повідомлення, яке потрібно передати в обох видах мереж, становить 200 Кбайт. Відправник знаходиться на відстані 5000 км від отримувача. Пропускна здатність ліній зв'язку становить 2 Мбіт/с.

Час передачі даних по мережі з комутацією каналів складається з часу поширення сигналу, що для відстані 5000 км можна оцінити приблизно в 25 мс, і часу передачі повідомлення, що при пропускній здатності 2 Мбіт/с і довжині повідомлення 200 Кбайт дорівнює приблизно 800 мс, тобто разом передача даних зайняла 825 мс.

Оцінимо додатковий час, потрібний для передачі цього повідомлення по мережі з комутацією пакетів. Вважатимемо, що шлях від відправника до отримувача пролягає через 10 комутаторів. Вихідне повідомлення розбивається на пакети в 1 Кбайт, усього 200 пакетів. Насамперед оцінимо затримку, яка виникає у вихідному вузлі. Припустимо, що частка службової інформації, розміщеної у заголовках пакетів, становить 10% від загального обсягу повідомлення. Отже, додаткова затримка, пов'язана з передачею заголовків пакетів, становить 10% від часу передачі цілого повідомлення, тобто 80 мс. Якщо прийняти інтервал між відправленням пакетів 1 мс, тоді додаткові втрати за рахунок інтервалів складуть 200 мс. Загалом, у вихідному вузлі через пакетування повідомлення під час передачі виникла додаткова затримка в 280 мс.

Кожен з 10 комутаторів вносить затримку комутації, яка може мати великий розкид, від частки до тисяч мілісекунд. У даному прикладі приймемо, що на комутацію в середньому витрачається 20 мс. Крім того, під час проходження повідомлень через комутатор виникає затримка буферизації пакета. Ця затримка при розмірі пакета 1 Кбайт і пропускній здатності лінії 2 Мбіт/с дорівнює 4 мс. Загальна затримка, внесена 10 комутаторами, складе приблизно 240 мс. У результаті додаткова затримка, створена мережею з комутацією пакетів, склала 520 мс. З огляду на те, що вся передача даних у мережі з комутацією каналів зайняла 825 мс, цю додаткову затримку можна вважати суттєвою.

Хоча наведений розрахунок носить дуже приблизний характер, він робить зрозумілішими причини, які призводять до того, що процес передачі для певної пари абонентів у мережі з комутацією пакетів є повільнішим, ніж у мережі з комутацією каналів.

Невизначена пропускна здатність мережі з комутацією пакетів – це плата за її загальну ефективність при деякому обмеженні інтересів окремих абонентів. Аналогічно, у мультипрограмній операційній системі час виконання додатка передбачити заздалегідь неможливо, оскільки він залежить від кількості інших додатків, з якими даний додаток «поділяє» процесор.

На ефективність роботи мережі значно впливають розміри пакетів, які передає мережа. Занадто великі розміри пакетів наближають мережу з комутацією пакетів до мережі з комутацією каналів, тому ефективність мережі при цьому падає. Занадто малі пакети помітно збільшують частку службової інформації, оскільки кожен пакет несе з собою заголовок фіксованої довжини, а кількість пакетів, на які розбиваються повідомлення, різко зростатиме при зменшенні розміру пакета. Існує деяка золота середина, що забезпечує максимальну ефективність роботи мережі, однак її важко визначити точно, оскільки вона залежить від багатьох факторів, деякі з них до того ж постійно змінюються в процесі роботи мережі. Тому розробники протоколів для мереж з комутацією пакетів вибирають межі, у яких може перебувати довжина пакета, а точніше його поле даних, оскільки заголовок, як правило, має фіксовану довжину. Зазвичай нижня межа поля даних обирається такою, що дорівнює нулю, і це дозволяє передавати службові пакети без користувальницьких даних, а верхня межа не перевищує чотирьох кілобайт. Додатки під час передачі даних намагаються зайняти максимальний розмір поля даних, щоб швидше виконати обмін даними, а невеликі пакети зазвичай використовуються для квитанцій щодо доставки пакета.

Під час вибору розміру пакета необхідно враховувати також і інтенсивність бітових помилок каналу. На ненадійних каналах необхідно зменшувати розміри пакетів, оскільки це зменшує обсяг повторно переданих даних при перекручуваннях пакетів.

9.2 Комутація повідомлень

Під *комутацією повідомлень* розуміють передачу єдиного блоку даних між транзитними комп'ютерами мережі з часовою буферизацією цього блоку на диску кожного комп'ютера (рис. 9.2). Повідомлення, на відміну від пакета, має довільну довжину, що визначається не технологічними міркуваннями, а змістом інформації, яка складає повідомлення. Наприклад, повідомленням може бути текстовий документ, файл з кодом програми, електронний лист.

Транзитні комп'ютери можуть з'єднуватися між собою як мережею з комутацією пакетів, так і мережею з комутацією каналів. Повідомлення зберігається в транзитному комп'ютері на диску, причому час зберігання може бути досить великим, якщо комп'ютер завантажений іншими роботами або мережа тимчасово перевантажена.

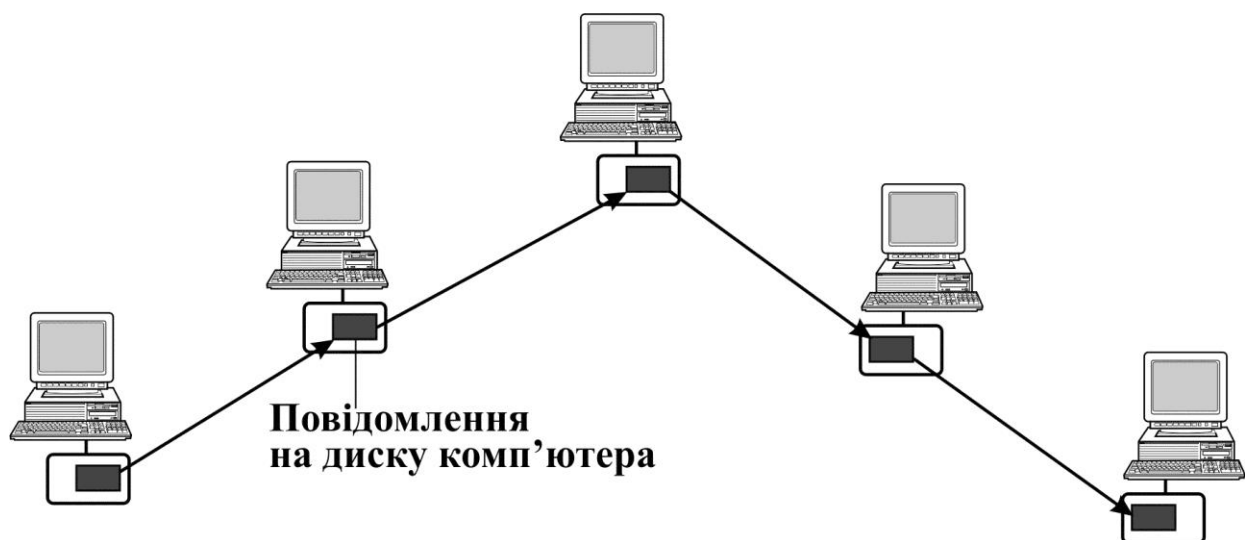


Рисунок 9.2 – Комутація повідомлень

За такою схемою зазвичай передаються повідомлення, які не вимагають негайної відповіді, найчастіше повідомлення електронної пошти. Режим передачі з проміжним зберіганням на диску називається режимом «зберігання-і-передача» (*store-and-forward*).

Режим комутації повідомлень розвантажує мережу для передачі трафіка, який потребує швидкої відповіді, наприклад, трафіка служби WWW або файлової служби.

Кількість транзитних комп'ютерів намагаються за можливості зменшити. Якщо комп'ютери підключені до мережі з комутацією пакетів, то кількість проміжних комп'ютерів зазвичай зменшується до двох. Наприклад, користувач передає поштове повідомлення своєму серверу вихідної пошти, а той відразу намагається передати повідомлення серверу вхідної пошти адресата. Але якщо комп'ютери пов'язані між собою телефонною мережею, то часто використовується кілька проміжних серверів, оскільки прямий доступ до кінцевого сервера може бути неможливий у цей момент через перевантаження телефонної мережі (абонент зайнятий) або економічно не вигідний через високі тарифи на далекий телефонний зв'язок.

Техніка комутації повідомлень з'явилася в комп'ютерних мережах раніше техніки комутації пакетів, але потім була витиснута останньою як ефективнішою за критерієм пропускної здатності мережі. Запис повідомлення на диск займає досить багато часу, крім того, наявність дисків припускає спеціалізовані комп'ютери як комутатори, що здорожує мережу.

Сьогодні комутація повідомлень працює тільки для деяких не оперативних служб, причому найчастіше поверх мережі з комутацією пакетів, як служба прикладного рівня.

9.3 Передача з встановленням та без встановлення з'єднань

Під час передачі кадрів даних на каналному рівні використовуються як дейтаграмні процедури, які працюють без встановлення з'єднання (connectionless), так і процедури з попереднім встановленням логічного з'єднання (connection-oriented).

Під час дейтаграмної передачі кадр посилається у мережу «без попередження», і ніякої відповідальності за його втрату протокол не несе (рис. 9.3, а). Припускається, що мережа завжди готова прийняти кадр від кінцевого вузла. Дейтаграмний метод працює швидко, оскільки ніяких попередніх дій перед відправленням даних не виконується. Однак під час такого методу важко організувати в рамках протоколу відстеження факту доставки кадра вузлу призначення. Цей метод не гарантує доставку пакета.

Передача зі встановленням з'єднання надійніша, але вимагає більше часу для передачі даних і обчислювальних витрат від кінцевих вузлів.

У цьому випадку вузлу-отримувачу надсилається службовий кадр спеціального формату з пропозицією встановити з'єднання (рис. 9.3, б).

Якщо вузол-отримувач згоден з цим, то він посилає у відповідь інший службовий кадр, який підтверджує встановлення з'єднання, й пропонує для даного логічного з'єднання деякі параметри, наприклад ідентифікатор з'єднання, максимальне значення поля даних кадрів, які використовуватимуться в рамках даного з'єднання тощо. Вузол-ініціатор з'єднання може завершити процес установа з'єднання відправленням третього службового кадра, у якому повідомить, що запропоновані параметри йому підходять. На цьому логічне з'єднання вважається встановленим, і в його рамках можна передавати інформаційні кадри з користувальницькими даними. Після передачі деякого закінченого набору даних, наприклад, певного файлу, вузол ініціює розрив даного логічного з'єднання, посилаючи відповідний службовий кадр.

Зазначимо, що, на відміну від протоколів дейтаграмного типу, які підтримують тільки один тип кадра – інформаційний, протоколи, що працюють по процедурі із установа з'єднання, мають підтримувати кілька типів кадрів – службові, для встановлення (і розриву) з'єднання, і інформаційні, що переносять властиво користувальницькі дані.

Логічне з'єднання забезпечує передачу даних як в одному напрямі – від ініціатора з'єднання, так і в обох напрямках.

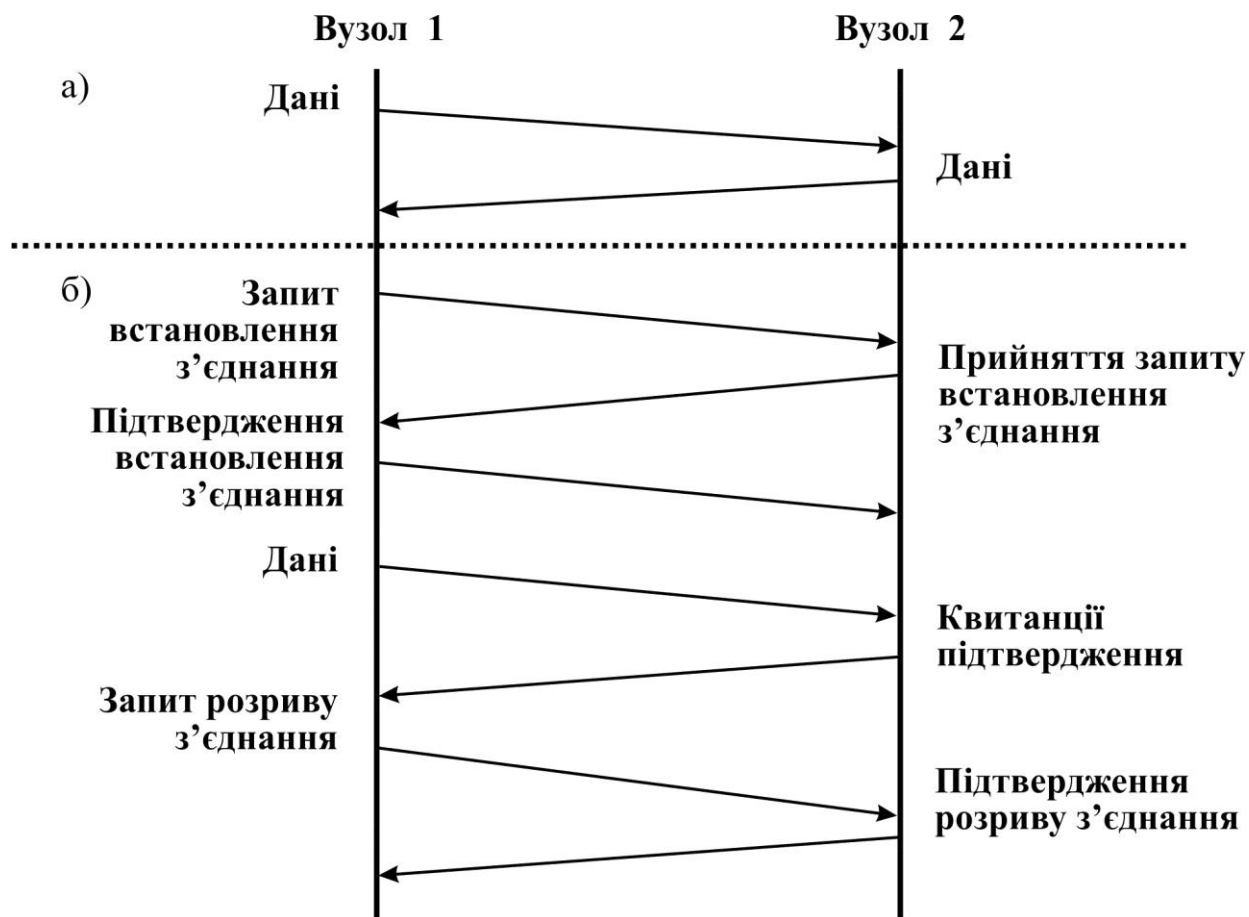


Рисунок 9.3 – Протоколи без встановлення з'єднання (а) і із встановленням з'єднання (б)

Процедура встановлення з'єднання може використовуватися для досягнення різних цілей:

- для взаємної аутентифікації користувачів або обладнання (маршрутизатори теж можуть мати імена й паролі, які потрібні для впевненості у тому, що зломисник не підмінив корпоративний маршрутизатор і не відвів потік даних у свою мережу для аналізу);
- для узгодження змінюваних параметрів протоколу: MTU, різних тайм-аутів тощо;
- для виявлення й корекції помилок. Встановлення логічного з'єднання дає точку відліку для завдання початкових значень номерів кадрів. При втраті нумерованого кадру приймач, по-перше, отримує можливість виявити цей факт, а по-друге, він може повідомити передавачу, який у точності кадр потрібно передати повторно;
- у деяких технологіях процедуру встановлення логічного з'єднання використовують при динамічному настроюванні комутаторів мережі для маршрутизації всіх наступних кадрів, які проходять через мережу

в рамках даного логічного з'єднання. Так працюють мережі технологій X.25, frame relay і ATM.

Як видно з наведеного списку, при встановленні з'єднання можуть переслідуватися різні цілі, у деяких випадках – дещо одночасно.

9.4 Контрольні запитання та завдання

1. Поясніть відмінність між трьома поняттями:
 - логічні з'єднання, на яких засновані деякі протоколи;
 - віртуальні канали в мережах з комутацією пакетів;
 - складені канали в мережах з комутацією каналів.
2. Назвіть, який тип кадрів підтримує протокол дейтаграмного типу?
3. Які типи кадрів підтримують протоколи, що працюють за процедурою з установленням з'єднання?
4. З яких міркувань для мереж з комутацією пакетів обирають межі, у яких може перебувати довжина пакета?
5. Як устатковується динамічний віртуальний канал?

Лекція 10

МЕТОДИ ПОВТОРНОЇ ПЕРЕДАЧІ ДАНИХ

10.1 Методи забезпечення достовірної передачі пакетів у мережі

Канальний рівень має виявляти помилки передачі даних, пов'язані з перекручуванням біт у прийнятому кадрі даних або з втратою кадру, і, за можливості, їх коректигувати.

Більша частина протоколів канального рівня виконує лише першу задачу – виявлення помилок, вважаючи, що коригувати помилки, тобто повторно передавати дані, які містили перекручену інформацію, мають протоколи верхніх рівнів. Так працюють такі популярні протоколи локальних мереж, як Ethernet, Token Ring, FDDI та інших. Однак існують протоколи канального рівня, наприклад, LLC2 або LAP-B, які самостійно вирішують задачу відновлення перекручених або загублених кадрів.

Вочевидь, що протоколи мають працювати найефективніше в типових умовах роботи мережі. Тому для мереж, у яких перекручування й втрата кадрів є дуже рідкісними подіями, розробляються протоколи типу Ethernet, у яких не передбачаються процедури усунення помилок. Дійсно, наявність процедур відновлення даних вимагало б від кінцевих вузлів додаткових обчислювальних витрат, які в умовах надійної роботи мережі були б надлишковими.

Навпаки, якщо в мережі перекручування та втрати відбуваються часто, то бажано вже на канальному рівні використовувати протокол з корекцією помилок, а не залишати цю роботу протоколам верхніх рівнів. Протоколи верхніх рівнів, наприклад, транспортного або прикладного, працюючи з великими тайм-аутами, відновлять загублені дані з великою затримкою. У глобальних мережах перших поколінь, наприклад, мережах X.25, які працювали через ненадійні канали зв'язку, протоколи канального рівня завжди виконували процедури відновлення загублених і перекручених кадрів.

Тому не можна вважати, що один протокол краще іншого тому, що він відновлює помилкові кадри, а другий – ні. Кожен протокол має працювати в тих умовах, для яких він розроблений.

Методи виявлення помилок

Усі методи виявлення помилок ґрунтуються на передачі в складі кадра даних службової надлишкової інформації, за якою можна з деяким ступенем

імовірності робити висновки про вірогідність прийнятих даних. Цю службову інформацію прийнято називати *контрольною сумою* або *послідовністю контролю кадра* (*Frame Check Sequence, FCS*). Контрольна сума обчислюється як функція від основної інформації, причому, необов'язково тільки шляхом підсумовування. Приймаюча сторона вдруге обчислює контрольну суму кадра за відомим алгоритмом і у випадку її збігу з контрольною сумою, обчисленою передавальною стороною, робить висновок про те, що дані були передані через мережу коректно.

Існує кілька розповсюджених алгоритмів обчислення контрольної суми, які відрізняються обчислювальною складністю й здатністю виявляти помилки в даних.

Контроль за паритетом є найпростішим методом контролю даних. У той самий час це найменш потужний алгоритм контролю, оскільки з його допомогою можна виявити лише одиночні помилки у даних, що перевіряються. Метод полягає в підсумовуванні за модулем 2 всіх біт контрольованої інформації. Наприклад, для даних 100101011 результатом контрольного підсумовування буде значення 1. Результат підсумовування також є одним бітом даних, який пересилається разом з контрольованою інформацією. При перекручуванні під час пересилання будь-якого одного біта вихідних даних (або контрольного розряду) результат підсумовування відрізнятиметься від прийнятого контрольного розряду, що свідчить про помилку. Однак подвійна помилка, наприклад, 110101010, буде неправильно прийнята за коректні дані. Тому контроль за паритетом застосовується до невеликих порцій даних, як правило, до кожного байта, що дає коефіцієнт надмірності для цього методу 1/8. Метод рідко застосовується в обчислювальних мережах через його велику надмірність і невисокі діагностичні можливості.

Вертикальний і горизонтальний контроль за паритетом є модифікацією описаного вище методу. Його відмінність у тому, що вихідні дані розглядаються у вигляді матриці, рядки якої становлять байти даних. Контрольний розряд підраховується окремо для кожного рядка й для кожного стовпця матриці. Цей метод виявляє велику частину подвійних помилок, однак має ще більшу надмірність. На практиці сьогодні також майже не застосовується.

Циклічний надлишковий контроль (*Cyclic Redundancy Check, CRC*) є на сьогодні найпопулярнішим методом контролю в обчислювальних мережах (і не тільки в мережах, наприклад, цей метод широко застосовується під час запису даних на диски). Метод базується на поданні вихідних даних у вигляді одного багаторозрядного двійкового числа. Наприклад, кадр стандарту Ethernet,

що складає 1024 байт, розглядатиметься як одне число, що становить 8192 біт. Як контрольна інформація розглядається остача від ділення цього числа на відомий дільник R . Зазвичай як дільник обирається сімнадцяти- або тридцятитрирозрядне число таке, щоб остача від ділення мала довжину 16 розрядів (2 байти) або 32 розряду (4 байти). При отриманні кадра даних знову обчислюється остача від ділення на той же дільник R , але при цьому до даних кадра додається й контрольна сума, що міститься в ньому. Якщо остача від ділення на R дорівнює нулю (існує дещо модифікована процедура обчислення остачі, яка призводить до отримання у випадку відсутності помилок відомої ненульової остачі, що є надійнішим показником коректності), то робиться висновок про відсутність помилок в отриманому кадрі, у протилежному випадку кадр вважається перекрученим.

Цей метод має більш високу обчислювальну складність, але його діагностичні можливості набагато вищі, ніж у методів контролю за паритетом. Метод CRC виявляє всі поодинокі помилки, подвійні помилки й помилки в непарній кількості біт. Метод має також невисокий ступінь надмірності. Наприклад, для кадра Ethernet розміром в 1024 байт контрольна інформація довжиною в 4 байт становить лише 0,4 %.

Методи відновлення перекручених і загублених кадрів

Методи корекції помилок в обчислювальних мережах базуються на повторній передачі кадра даних у тому випадку, якщо кадр губиться й не доходить до адресата або приймач виявив у ньому перекручування інформації. Щоб переконатися в необхідності повторної передачі даних, відправник нумерує кадри, що відправляються, і для кожного кадра очікує від приймача так званої *позитивної квитанції* – службового кадра, який сповіщає про те, що вихідний кадр був отриманий і дані в ньому виявилися коректними. Час цього очікування обмежений – при відправленні кожного кадра передавач запускає таймер і, якщо по закінченні часу позитивна квитанція не отримана, кадр вважається загубленим. Приймач у випадку отримання кадра з перекрученими даними може відправити *негативну квитанцію* – явне зазначення того, що даний кадр потрібно передати повторно.

Існують два підходи до організації процесу обміну квитанціями: з простоями і з організацією «вікна».

10.2 Метод передачі з зупинкою та очікуванням

Метод із простоями (*Idle Source*) вимагає, щоб джерело, яке послало кадр, очікувало отримання квитанції (позитивної або негативної) від приймача й тільки після цього посилало наступний кадр (або повторювало перекручений). Якщо ж квитанція не приходить протягом тайм-ауту, то кадр (або квитанція) вважається загубленим і його передача повторюється. На рис. 10.1, а видно, що в цьому випадку продуктивність обміну даними істотно знижується, – хоча передавач і міг би послати наступний кадр відразу ж після відправлення попереднього, він зобов’язаний очікувати приходу квитанції. Зниження продуктивності цього методу корекції особливо помітно на низькошвидкісних каналах зв’язку, тобто в територіальних мережах.

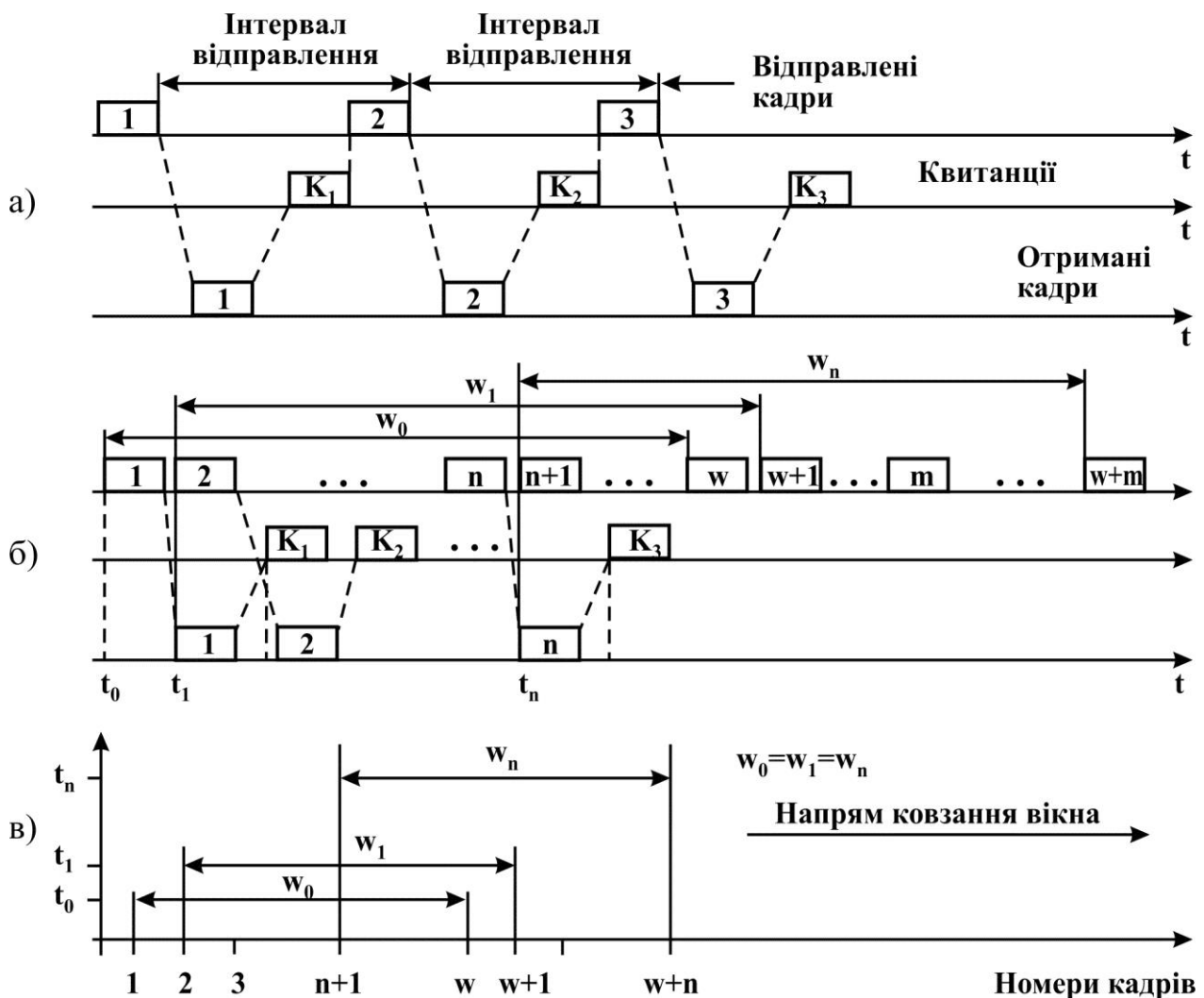


Рисунок 10.1 – Методи відновлення перекручених і загублених кадрів

Другий метод називається методом «ковзаючого вікна» (*sliding window*). Тут для підвищення коефіцієнта використання лінії джерелу дозволяється передати деяку кількість кадрів у безперервному режимі, тобто в максимально можливому для джерела темпі, без отримання на ці кадри позитивних відповідних квитанцій. (Далі, де це не перекручує суть розглянутого питання, позитивні квитанції для стислості називатимуться просто «квитанціями».) Кількість кадрів, які дозволяється так передавати, називається розміром вікна. Рисунок 10.1, б ілюструє даний метод для вікна розміром у W кадрів.

У початковий момент, коли ще не послано жодного кадру, вікно визначає діапазон кадрів з номерами від 1 до W включно. Джерело починає передавати кадри й отримувати у відповідь квитанції. Для простоти припустимо, що квитанції надходять у тій же послідовності, що й кадри, яким вони відповідають. У момент t_1 при отриманні першої квитанції K_1 вікно зсувається на одну позицію, визначаючи новий діапазон від 2 до $(W + 1)$.

Процеси відправлення кадрів і отримання квитанцій йдуть досить незалежно один від одного. Розглянемо довільний момент часу t_n , коли джерело отримало квитанцію на кадр з номером n . Вікно зсунулося вправо й визначило діапазон дозволених до передачі кадрів від $(n + 1)$ до $(W + n)$.

Усю безліч кадрів, що виходять із джерела, можна розділити на зазначені нижче групи (рис. 10.1, б):

- *Кадри з номерами від 1 до n* , уже були відправлені й квитанції на них отримані, тобто вони знаходяться за межами вікна зліва.
- *Кадри, починаючи з номера $(n + 1)$ і закінчуючи номером $(W + n)$* , знаходяться у межах вікна й тому можуть бути відправлені без очікування приходу будь-якої квитанції. Цей діапазон може розділятися ще на два піддіапазони:
 - кадри з номерами від $(n + 1)$ до m , які вже відправлені, але квитанції на них ще не отримані;
 - кадри з номерами від m до $(W + n)$, які поки не відправлені, хоча заборони на це немає.
- *Всі кадри з номерами, більшими або такими, що дорівнюють $(W + n + 1)$* , знаходяться за межами вікна справа і тому поки не можуть бути відправлені.

Переміщення вікна уздовж послідовності номерів кадрів показано на рис. 10.1, в. Тут t_0 – вихідний момент, t_1 і t_n – моменти приходу квитанцій на 1-й і n -й кадр відповідно. Щоразу, коли надходить квитанція, вікно

зсувається вліво, але його розмір при цьому не змінюється й дорівнює W . Зазначимо, що хоча у даному прикладі розмір вікна у процесі передачі залишається постійним, у реальних протоколах (наприклад, TCP) можна зустріти варіанти даного алгоритму з розміром вікна, що змінюється.

Отже, під час відправлення кадра з номером n джерелу дозволяється передати ще $(W - 1)$ кадрів до отримання квитанції на кадр n , так що в мережу останнім піде кадр з номером $(W + n - 1)$. Якщо ж за цей час квитанція на кадр n так і не прийшла, то процес передачі припиняється, і після закінчення деякого тайм-ауту кадр n (або квитанція на нього) вважається загубленим, і він передається знову.

Якщо ж потік квитанцій надходить більш-менш регулярно, у межах допуску в W кадрів, то швидкість обміну досягає максимально можливої величини для даного каналу й прийнятого протоколу.

Метод ковзаючого вікна складніший у реалізації, ніж метод з простоями, оскільки передавач має зберігати в буфері всі кадри, на які ще не отримані позитивні квитанції. Крім того, потрібно відслідковувати кілька параметрів алгоритму: розмір вікна W , номер кадра, на який отримана квитанція, номер кадра, який ще можна передати до отримання нової квитанції.

Приймач може не посилати квитанції на кожен прийнятий коректний кадр. Якщо декілька кадрів прийшли майже водночас, то приймач може послати квитанцію тільки на останній кадр. При цьому мається на увазі, що всі попередні кадри також дійшли успішно.

Деякі методи використовують негативні квитанції. Негативні квитанції бувають двох типів – групові й вибіркові. Групова квитанція містить номер кадра, починаючи з якого слід повторити передачу всіх кадрів, відправлених передавачем у мережу. Вибіркова негативна квитанція вимагає повторної передачі тільки одного кадра.

Метод ковзаючого вікна реалізований у багатьох протоколах: LLC2, LAP-B, X.25, TCP, Novell NCP Burst Mode.

Метод з простоями є окремим випадком методу ковзаючого вікна, коли розмір вікна дорівнює одиниці.

Метод ковзаючого вікна має два параметри, які можуть помітно впливати на ефективність передачі даних між передавачем та приймачем, – розмір вікна й величину тайм-ауту очікування квитанції. У надійних мережах, коли кадри перекручуються й губляться рідко, для підвищення швидкості обміну даними розмір вікна потрібно збільшувати, оскільки тоді передавач посилатиме кадри з меншими паузами. У ненадійних мережах розмір вікна слід зменшувати,

оскільки через часті втрати й перекручування кадрів різко зростає обсяг вдруге переданих через мережу кадрів, а, значить, пропускна здатність мережі витрачатиметься багато в чому вхолосту – корисна пропускна здатність мережі падатиме.

Вибір тайм-ауту залежить не від надійності мережі, а від затримок передачі кадрів мережею.

У багатьох реалізаціях методу ковзаючого вікна величина вікна й тайм-аут обираються адаптивно, залежно від поточного стану мережі.

10.3 Контрольні запитання та завдання

1. Як передавач визначає факт втрати позитивної квитанції у методі ковзаючого вікна?

2. Мережа з комутацією пакетів зазнає перевантаження. Для усунення цієї ситуації розмір вікна в протоколах комп'ютерів мережі необхідно збільшити чи зменшити?

3. Як впливає надійність ліній зв'язку в мережі на вибір розміру вікна?

Лекція 11

ПРИНЦИПИ ПОБУДОВИ МЕРЕЖНОГО РІВНЯ

11.1 Побудова мережі на основі протоколів мережного рівня

Принципи об'єднання мереж на основі протоколів мережного рівня

У стандартній моделі взаємодії відкритих систем у функції мережного рівня входить вирішення таких задач:

- передача пакетів між кінцевими вузлами в складених мережах;
- вибір маршруту передачі пакетів, який буде найкращим за деяким критерієм;
- узгодження різних протоколів канального рівня, які використовуються в окремих підмережах однієї складеної мережі.

Протоколи мережного рівня реалізуються, як правило, у вигляді програмних модулів і виконуються на кінцевих вузлах-комп'ютерах, які називаються хостами, а також на проміжних вузлах-маршрутизаторах, що називаються шлюзами. Функції маршрутизаторів можуть виконувати як спеціалізовані пристрої, так і універсальні комп'ютери з відповідним програмним забезпеченням.

Обмеження мостів і комутаторів

Створення складної, структурованої мережі, яка інтегрує різні базові технології, може здійснюватися й засобами канального рівня, – для цього можуть бути використані деякі типи мостів і комутаторів. Міст або комутатор розділяє мережу на сегменти, локалізуючи трафік усередині сегмента, що робить лінії зв'язку поділюваними переважно між станціями даного сегмента. Тим самим мережа розпадається на окремі підмережі, з яких можуть бути побудовані складені мережі досить великих розмірів.

Але побудова складних мереж тільки на основі повторювачів, мостів і комутаторів має суттєві обмеження й недоліки:

- *По-перше*, у топології отриманої мережі мають бути відсутніми петлі. Дійсно, міст/комутатор може вирішувати задачу доставки пакета адресату тільки тоді, коли між відправником і отримувачем існує єдиний шлях. Водночас наявність надлишкових зв'язків, які й утворюють петлі, часто необхідна для кращого балансування навантаження, а також для підвищення надійності мережі за рахунок утворення резервних шляхів.

- **По-друге**, логічні сегменти мережі, що розташовані між мостами або комутаторами, *слабко ізольовані* один від одного, а саме – не захищені від так званих ширококомовних штормів. Якщо будь-яка станція посилає ширококомовне повідомлення, то це повідомлення передається всім станціям всіх логічних сегментів мережі. Захист від ширококомовних штормів у мережах, побудованих на основі мостів і комутаторів, має кількісний, а не якісний характер: адміністратор просто обмежує кількість ширококомовних пакетів, що дозволяється генерувати деякому вузлу в одиницю часу. Використання ж механізму віртуальних мереж, реалізованого в багатьох комутаторах, хоча й дозволяє досить гнучко створювати ізольовані по трафіку групи станцій, але при цьому ізолює їх повністю, так що вузли однієї віртуальної мережі не можуть взаємодіяти з вузлами іншої віртуальної мережі.

- **По-третьє**, в мережах, побудованих на основі мостів і комутаторів, досить складно вирішується задача управління трафіком на основі вагомості даних, що містяться в пакеті. У таких мережах це можливо тільки за допомогою користувальницьких фільтрів, для завдання яких адміністратору доводиться мати справу з двійковим поданням вмісту пакетів.

- **По-четверте**, реалізація транспортної підсистеми тільки засобами фізичного й каналного рівнів, до яких належать мости й комутатори, призводить до недостатньо гнучкої, однорівневої системи адресації: як адреса призначення використовується MAC-адреса, жорстко пов'язана з мережним адаптером.

- **Нарешті**, можливістю трансляції протоколів каналного рівня володіють далеко не всі типи мостів і комутаторів, до того ж ці можливості обмежені. Зокрема, у поєднаних мережах мають збігатися максимально припустимі розміри полів даних у кадрах, оскільки мостами й комутаторами не підтримується функція фрагментації кадрів. Наявність серйозних обмежень у протоколів каналного рівня показує, що побудова на основі засобів цього рівня великих неоднорідних мереж є досить проблематичною. Природне вирішення в цих випадках – це залучення засобів більш високого, мережного рівня.

Поняття internetworking

Основна ідея введення мережного рівня полягає в такому. Мережа в загальному випадку розглядається як сукупність декількох мереж і називається складеною мережею або інтермережею (*internetwork* або *internet*). Мережі, що входять у складену мережу, називаються підмережами (*subnet*), *складовими* мережами або просто мережами (рис. 11.1).

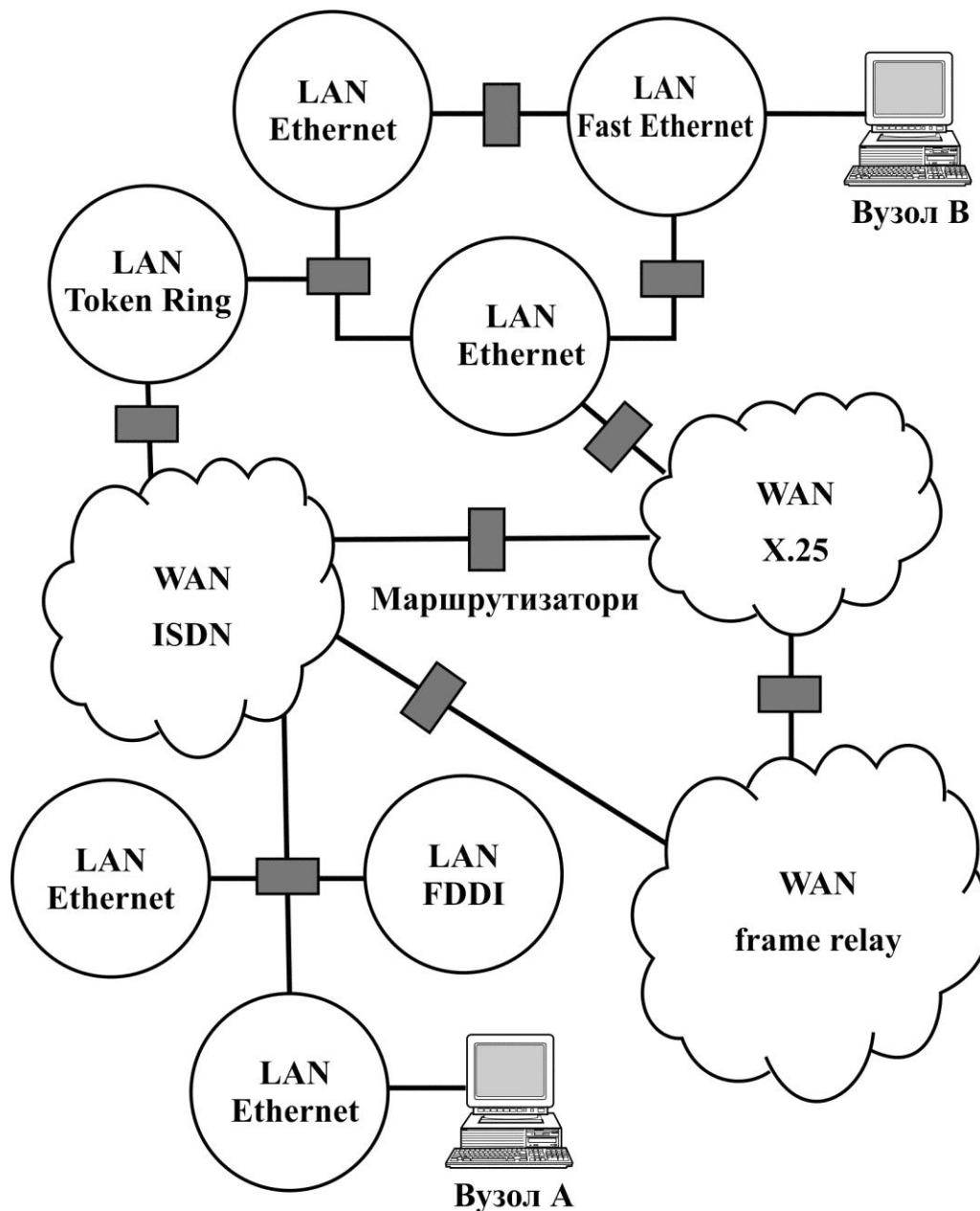


Рисунок 11.1 – Архітектура складеної мережі

Підмережі з'єднуються між собою маршрутизаторами. Компонентами складеної мережі можуть бути як локальні, так і глобальні мережі. Внутрішня структура кожної мережі на рисунку не показана, оскільки вона не має значення при розгляді мережного протоколу. Всі вузли в межах однієї підмережі взаємодіють, використовуючи єдину для них технологію. Так, у складену мережу, показану на рисунку, входить кілька мереж різних технологій: локальні мережі Ethernet, Fast Ethernet, Token Ring, FDDI і глобальні мережі frame relay, X.25, ISDN. Кожна з цих технологій достатня для того, щоб організувати взаємодію всіх вузлів у своїй підмережі, але не спроможна побудувати інформаційний зв'язок між довільно обраними вузлами, які належать різним підмережам,

наприклад, між вузлом А і вузлом В на рис. 11.1. Отже, для організації взаємодії між будь-якою довільною парою вузлів цієї «великої» складеної мережі потрібні додаткові засоби. Такі засоби й надає мережний рівень.

Мережний рівень виступає як координатор, що організує роботу всіх підмереж, що лежать на шляху просування пакета по складеній мережі. Для переміщення даних у межах підмереж мережний рівень звертається до використовуваних в цих підмережах технологій.

Хоча багато технологій локальних мереж (Ethernet, Token Ring, FDDI, Fast Ethernet і ін.) використовують ту саму систему адресації вузлів на основі MAC-адрес, існує чимало технологій (X.25, ATM, frame relay), у яких застосовуються інші схеми адресації. Адреси, що присвоєні вузлам відповідно до технологій підмереж, називають локальними. Щоб мережний рівень міг виконати свою задачу, йому необхідна власна система адресації, яка не залежить від способів адресації вузлів у окремих підмережах, що дозволила б на мережному рівні універсальним і однозначним способами ідентифікувати будь-який вузол складеної мережі.

Природним способом формування мережної адреси є унікальна нумерація всіх підмереж складеної мережі й нумерація всіх вузлів у межах кожної підмережі. Отже, мережна адреса є парою: номер мережі (підмережі) і номер вузла.

Як номер вузла може виступати або локальна адреса цього вузла (така схема прийнята у стеку IPX/SPX), або деяке число, ніяк не пов'язане з локальною технологією, яке однозначно ідентифікує вузол у межах даної підмережі. У першому випадку мережна адреса стає залежною від локальних технологій, що обмежує її застосування. Наприклад, мережні адреси IPX/SPX розраховані на роботу в складених мережах, що поєднують мережі, де використовуються лише MAC-адреси або адреси аналогічного формату. Другий підхід більш універсальний, він характерний для стека TCP/IP. І в тому і іншому випадку кожен вузол складеної мережі має поряд зі своєю локальною адресою ще одну – універсальну мережну адресу.

Дані, що надходять на мережний рівень і які необхідно передати через складену мережу, забезпечуються заголовком мережного рівня. Дані разом із заголовком утворюють пакет. Заголовок пакета мережного рівня має уніфікований формат, що не залежить від форматів кадрів канального рівня тих мереж, які можуть входити в об'єднану мережу, і несе поряд з іншою службовою інформацією дані про номер мережі, якій призначається цей пакет. Мережний рівень визначає маршрут і переміщує пакет між підмережами.

В ході передачі пакета з однієї підмережі в іншу пакет мережного рівня, інкапсульований у прибулий каналний кадр першої підмережі, звільняється від заголовків цього кадра й оточується заголовками кадра каналного рівня подальшої підмережі. Інформацією, на основі якої робиться ця заміна, є службові поля пакета мережного рівня. У полі адреси призначення нового кадра вказується локальна адреса подальшого маршрутизатора.

Якщо в підмережі доставка даних здійснюється засобами каналного й фізичного рівнів (як, наприклад, у стандартних локальних мережах), то пакети мережного рівня впаковуються в кадри каналного рівня. Якщо ж у будь-якій підмережі для транспортування повідомлень використовується технологія, заснована на стеках з великою кількістю рівнів, то пакети мережного рівня впаковуються у блоки переданих даних найвищого рівня підмережі.

Якщо проводити аналогію між взаємодією різномірних мереж і перепискою людей з різних країн, то мережна інформація – це загальноприйнятий індекс країни, доданий до адреси листа, написаний на одній з сотні мов земної кулі, наприклад на санскриті. І навіть якщо цей лист має пройти через безліч країн, поштові працівники яких не знають санскриту, зрозумілий їм індекс країни-адресата підкаже, через які проміжні країни краще передати лист, щоб він найкоротшим шляхом потрапив до Індії. А вже там працівники місцевих поштових відділень зможуть прочитати точну адресу, яка вказує місто, вулицю, будинок і індивідуума, і доставити лист адресату, оскільки адреса написана мовою й у формі, прийнятій в даній країні.

Основним полем заголовка мережного рівня є номер мережі-адресата. У розглянутих нами раніше протоколах локальних мереж такого поля в кадрах не передбачено – припускалося, що всі вузли належать одній мережі. Явна нумерація мереж дозволяє протоколам мережного рівня складати точну карту міжмережних зв'язків і обирати раціональні маршрути за будь-якої їхньої топології, у тому числі альтернативні маршрути, якщо вони є, що не вміють робити мости й комутатори.

Крім номера мережі заголовок мережного рівня має містити й іншу інформацію, необхідну для успішного переходу пакета з мережі одного типу в мережу іншого типу. До такої інформації може належати, наприклад:

- номер фрагмента пакета, необхідний для успішного проведення операцій збирання/розбирання фрагментів. Під час з'єднання мереж з різними максимальними розмірами пакетів;
- час життя пакета, що вказує, як довго він подорожує по інтермережі, цей час може використовуватися для знищення «заблудлих» пакетів;

- якість послуги – критерій вибору маршруту при міжмережних передачах, – наприклад, вузол-відправник може побажати передати пакет з максимальною надійністю, можливо, на шкоду часу доставки.

Коли дві або більше мережі організують спільну транспортну службу, то такий режим взаємодії зазвичай називають міжмережною взаємодією (internetworking).

11.2 Принципи маршрутизації

Найважливішою задачею мережного рівня є маршрутизація – передача пакетів між двома кінцевими вузлами в складеній мережі.

Розглянемо принципи маршрутизації на прикладі складеної мережі, зображеної на рис. 11.2.

У цій мережі 20 маршрутизаторів поєднують 18 мереж у загальну мережу; S1, S2, ... , S20 – це номери мереж. Маршрутизатори мають по декілька портів (принаймні, по два), до яких приєднуються мережі. Кожний порт маршрутизатора можна розглядати як окремий вузол мережі: він має власну мережну адресу й власна локальну адресу в тій підмережі, яка до нього підключена. Наприклад, маршрутизатор під номером 1 має три порти, до яких підключені мережі S1, S2, S3. На рисунку мережні адреси цих портів позначені як M1(1), M1(2) і M1(3). Порт M1(1) має локальну адресу в мережі з номером S1, порт M1(2) – у мережі S2, а порт M1(3) – у мережі S3. Отже, маршрутизатор можна розглядати як сукупність декількох вузлів, кожен з яких входить у свою мережу. Як єдиний пристрій маршрутизатор не має ні окремої мережної адреси, ні будь-якої локальної адреси.

Якщо маршрутизатор має блок управління (наприклад, SNMP-управління), то цей блок має власні локальну і мережну адреси, по яких до нього звертається центральна станція управління, що знаходиться десь у складеній мережі.

У складних складених мережах майже завжди існує кілька альтернативних маршрутів для передачі пакетів між двома кінцевими вузлами. Маршрут – це послідовність маршрутизаторів, які має пройти пакет від відправника до пункту призначення. Так пакет, відправлений з вузла А у вузол В, може пройти через маршрутизатори 17, 12, 5, 4 і 1 або маршрутизатори 17, 13, 7, 6 і 3. Неважко знайти ще кілька маршрутів між вузлами А і В.

Задачу вибору маршруту з декількох можливих вирішують маршрутизатори, а також кінцеві вузли. Маршрут обирається на підставі наявної в цих пристроїв інформації про поточну конфігурацію мережі, а також на підставі зазначеного

критерію вибору маршруту. Зазвичай критерієм виступає затримка проходження маршрута окремим пакетом або середньою пропускною здатністю маршрута для послідовності пакетів. Часто також використовується досить простий критерій, який враховує лише кількість пройдених у маршруті проміжних маршрутизаторів (хопів).

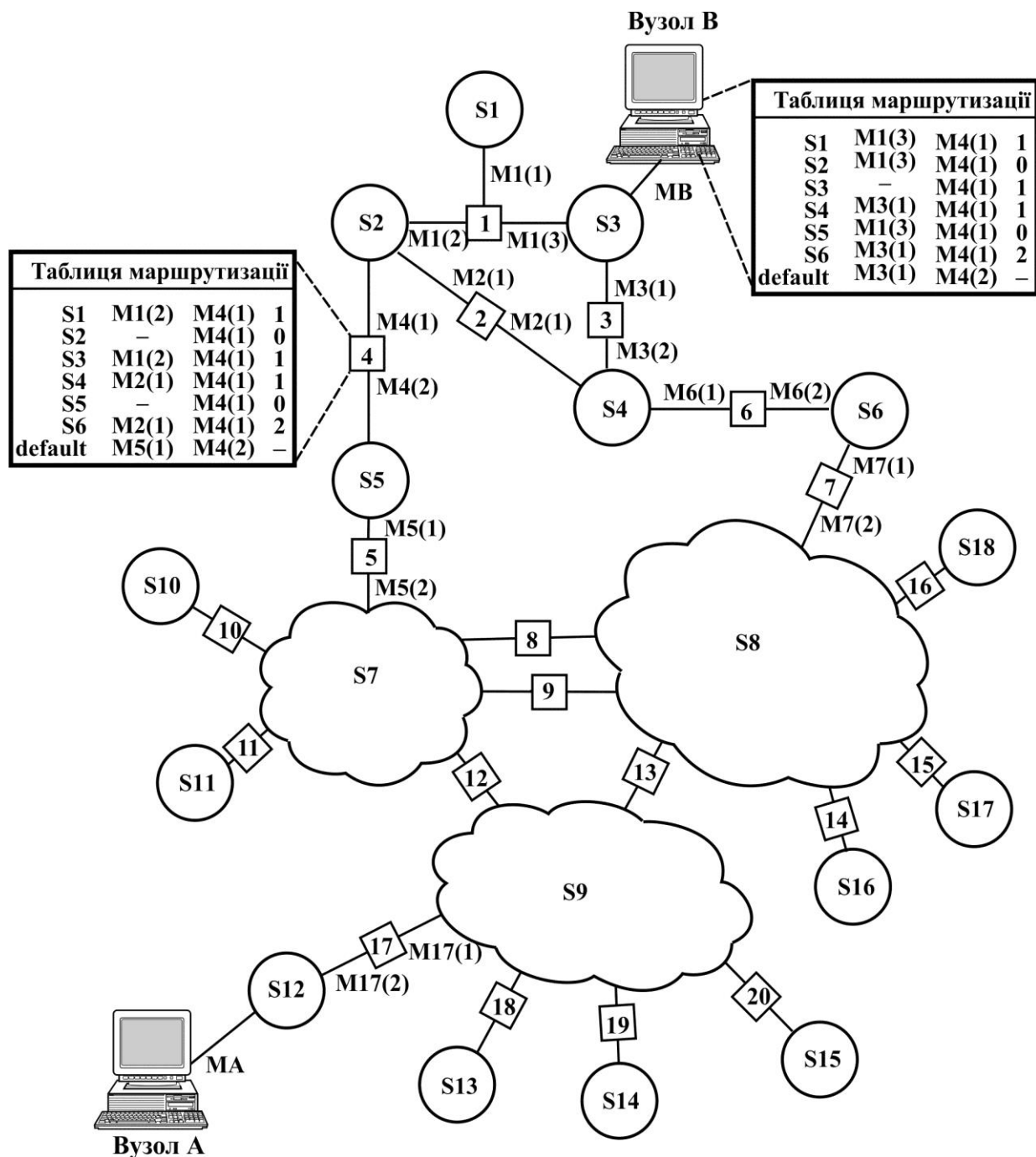


Рисунок 11.2 – Принципи маршрутизації в складеній мережі

Щоб за адресою мережі призначення можна було обрати раціональний маршрут подальшого проходження пакета, кожний кінцевий вузол і маршрутизатор аналізують спеціальну інформаційну структуру, яка називається

таблицею маршрутизації. Використовуючи умовні позначки для мережних адрес маршрутизаторів і номерів мереж у тому вигляді, як вони наведені на рис. 11.2, подивимося, як могла б виглядати таблиця маршрутизації, наприклад, у маршрутизаторі 4 (табл. 11.1).

Таблиця 11.1 – Таблиця маршрутизації маршрутизатора 4

Номер мережі призначення	Мережна адреса наступного маршрутизатора	Мережна адреса вихідного порту	Відстань до мережі призначення
S1	M1(2)	M4(1)	1
S2	–	M4(1)	0 (під'єднана)
S3	M1(2)	M4(1)	1
S4	M2(1)	M4(1)	1
S5	–	M4(2)	0 (під'єднана)
S6	M2(1)	M4(1)	2
Default	M5(1)	M4(2)	–

Таблиця 11.1 значно спрощена порівняно з реальними таблицями. Наприклад, відсутні стовпці з масками, ознаками стану маршруту, часом, протягом якого дійсні записи даної таблиці (їхнє застосування буде розглянуто пізніше). Крім того, як уже були сказано, тут зазначені адреси мереж умовного формату, які не відповідають будь-якому певному мережному протоколу. Проте ця таблиця містить основні поля, наявні в реальних таблицях в ході використання конкретних мережних протоколів, таких, як IP, IPX або X.25.

У першому стовпці таблиці перераховуються номери мереж, які входять в інтермережу. У кожному рядку таблиці услід за номером мережі вказується мережна адреса наступного маршрутизатора (точніше, мережна адреса відповідного порту наступного маршрутизатора), на який треба направити пакет, щоб той пересувався в напрямі мережі з даним номером раціональним маршрутом.

Коли на маршрутизатор надходить новий пакет, номер мережі призначення, витягнутий з кадра, що надійшов, послідовно порівнюється з номерами мереж з кожного рядка таблиці. Рядок з номером, що збігся, вказує мережі, на який найближчий маршрутизатор слід направити пакет. Наприклад, якщо на будь-який порт маршрутизатора 4 надходить пакет, адресований у мережу S6, то з таблиці маршрутизації випливає, що адреса наступного маршрутизатора – M2(1), тобто черговим етапом просування даного пакета буде рух до порту 1 маршрутизатора 2.

Оскільки пакет може бути адресований у будь-яку мережу складеної мережі, може здатися, що кожна таблиця маршрутизації повинна мати запис про всі мережі, що входять у складену мережу. Але при такому підході у випадку великої мережі обсяг таблиць маршрутизації може виявитися дуже великим, що вплине на час її перегляду, потребує багато місця для зберігання тощо. Тому на практиці кількість записів у таблиці маршрутизації намагаються зменшити за рахунок використання спеціального запису – «*маршрутизатор за замовчуванням*» (*default*). Справді, якщо прийняти до уваги топологію складеної мережі, то в таблицях маршрутизаторів, що перебувають на периферії складеної мережі, досить записати номери мереж, безпосередньо приєднаних до даного маршрутизатора або розташованих поблизу, на тупикових маршрутах. Про всі ж інші мережі можна зробити в таблиці єдиний запис, що вказує на маршрутизатор, через який пролягає шлях до всіх цих мереж. Такий маршрутизатор називається маршрутизатором за замовчуванням, а замість номера мережі у відповідному рядку міститься особливий запис, наприклад, *default*. У нашому прикладі таким маршрутизатором за замовчуванням для мережі S5 є маршрутизатор 5, точніше його порт M5(1). Це означає, що шлях з мережі S5 майже до всіх мереж великої складеної мережі пролягає через цей порт маршрутизатора.

Перед тим як передати пакет наступному маршрутизатору, поточний маршрутизатор має визначити, на який з декількох власних портів він має помістити даний пакет. Для цього слугує третій стовпець таблиці маршрутизації. Ще раз підкреслимо, що кожний порт ідентифікується власною мережною адресою.

Деякі реалізації мережних протоколів допускають наявність у таблиці маршрутизації одразу декількох рядків, що відповідають одній і тій самій адресі мережі призначення. У цьому випадку в ході вибору маршруту приймається до уваги стовпець «Відстань до мережі призначення». Тут під відстанню розуміється будь-яка метрика, використовувана відповідно до заданого в мережному пакеті критерію (який часто називається класом сервісу). Відстань може вимірятися хопами, часом проходження пакета по лініях зв'язку, будь-якою характеристикою надійності ліній зв'язку на даному маршруті або іншою величиною, що відбиває якість даного маршруту по відношенню до заданого критерію. Якщо маршрутизатор підтримує кілька класів сервісу пакетів, то таблиця маршрутів складається й застосовується окремо для кожного виду сервісу (критерію вибору маршруту).

У табл. 11.1 відстань між мережами вимірялася хопами. Відстань для мереж, підключених безпосередньо до портів маршрутизатора, тут приймається рівною 0, але у деяких реалізаціях відлік відстаней починається з 1.

Наявність декількох маршрутів до одного вузла уможлиблюють передачу трафіка до цього вузла паралельно по декількох каналах зв'язку, – це підвищує пропускну здатність і надійність мережі.

Задачу маршрутизації вирішують не тільки проміжні вузли-маршрутизатори, але й кінцеві вузли-комп'ютери. Засоби мережного рівня, встановлені на кінцевому вузлі, під час обробки пакета мають, насамперед, визначити, направляється він в іншу мережу чи адресований якомусь вузлу даної мережі. Якщо номер мережі призначення збігається з номером даної мережі, то для такого пакета не потрібно вирішувати задачу маршрутизації. Якщо ж номери мереж відправлення й призначення не збігаються, – маршрутизація потрібна. Таблиці маршрутизації кінцевих вузлів повністю аналогічні таблицям маршрутизації, що зберігаються на маршрутизаторах.

Звернемося знову до мережі, поданої на рис. 11.2. Таблиця маршрутизації для кінцевого вузла В могла б виглядати так, як показано в табл. 11.2. Тут MB – мережна адреса порту комп'ютера В. На підставі цієї таблиці кінцевий вузол В обирає, на який з двох наявних у локальній мережі S3 маршрутизаторів слід посилати той або інший пакет.

Таблиця 11.2 – Таблиця маршрутизації кінцевого вузла В

Номер мережі призначення	Мережна адреса наступного маршрутизатора	Мережна адреса вихідного порту	Відстань до мережі призначення
S1	M1(3)	MB	1
S2	M1(3)	MB	1
S3	–	MB	0
S4	M3(1)	MB	1
S5	M1(3)	MB	2
S6	M3(1)	MB	2
Default	M3(1)	MB	–

Кінцеві вузли у ще більшому ступені, ніж маршрутизатори, користуються прийомом маршрутизації за замовчуванням. Хоча вони також у загальному випадку мають у своєму розпорядженні таблицю маршрутизації, її обсяг зазвичай незначний, що пояснюється периферійним розташуванням усіх кінцевих вузлів. Кінцевий вузол часто взагалі працює без таблиці маршрутизації, маючи

тільки відомості про адресу маршрутизатора за замовчуванням. За наявності одного маршрутизатора в локальній мережі цей варіант – єдино можливий для всіх кінцевих вузлів. Але навіть за наявності декількох маршрутизаторів у локальній мережі, коли перед кінцевим вузлом виникає проблема їх вибору, завдання маршруту за замовчуванням часто використовується у комп'ютерах для скорочення обсягу їхньої таблиці маршрутизації.

Таблиця 11.3 – це таблиця маршрутизації іншого кінцевого вузла складеної мережі – вузла А. Компактний вигляд таблиці маршрутизації відбиває той факт, що всі пакети, якію направляються з вузла А, або не виходять за межі мережі S12, або неодмінно проходять через порт 1 маршрутизатора 17. Саме він і визначений у таблиці маршрутизації як маршрутизатор за замовчуванням.

Таблиця 11.3 – Таблиця маршрутизації кінцевого вузла А

Номер мережі призначення	Мережна адреса наступного маршрутизатора	Мережна адреса вихідного порту	Відстань до мережі призначення
S12	–	MA	0
Default	M17	MA	–

Ще однією відмінністю роботи маршрутизатора й кінцевого вузла під час вибору маршруту є спосіб побудови таблиці маршрутизації. Якщо маршрутизатори зазвичай автоматично створюють таблиці маршрутизації, обмінюючись службовою інформацією, то для кінцевих вузлів таблиці маршрутизації часто створюються вручну адміністраторами та зберігаються у вигляді постійних файлів на дисках.

11.3 Контрольні запитання та завдання

1. У чому відмінність задач, розв'язуваних протоколами мережного рівня у локальних і глобальних мережах?

2. Порівняйте таблицю моста/комутатора з таблицею маршрутизатора. Як вони формуються? Яку інформацію містять? Від чого залежить їхній обсяг?

3. Таблиця маршрутизації містить записи про мережі призначення. Чи має вона містити записи про всі мережі складеної мережі або лише про деякі? Якщо лише про деякі, то про які саме?

4. Чи може в таблиці маршрутизації бути кілька записів про маршрутизатори за замовчуванням?

Лекція 12

ПРОТОКОЛИ МАРШРУТИЗАЦІЇ

12.1 Класифікація алгоритмів побудови таблиць маршрутизації

Задача маршрутизації вирішується на основі аналізу таблиць маршрутизації, розміщених у всіх маршрутизаторах і кінцевих вузлах мережі. Як саме відбувається формування цих таблиць? Якими засобами забезпечується адекватність інформації, яка міститься в них, щодо структури мережі, що постійно змінюється? Основна робота зі створення таблиць маршрутизації виконується автоматично, але й можливість вручну скорегувати або доповнити таблицю теж, як правило, передбачається.

Для автоматичної побудови таблиць маршрутизації маршрутизатори обмінюються інформацією про топологію складеної мережі відповідно до спеціального службового протоколу. Протоколи такого типу називаються протоколами маршрутизації (або маршрутизуючими протоколами). Протоколи маршрутизації (наприклад, RIP, OSPF, NLSP) слід відрізняти від власне мережних протоколів (таких, як IP, IPX). І ті й інші виконують функції мережного рівня моделі OSI – беруть участь у доставці пакетів адресату через неоднорідну складену мережу. Але в той час, як перші збирають і передають по мережі виключно службову інформацію, другі призначені для передачі користувальницьких даних, як це роблять протоколи канального рівня. Протоколи маршрутизації використовують мережні протоколи як транспортний засіб. Під час обміну маршрутною інформацією пакети протоколу маршрутизації поміщаються в поле даних пакетів мережного або навіть транспортного рівня, тому з огляду вкладеності пакетів протоколи маршрутизації формально слід було б віднести до більш високого рівня, ніж мережний.

У тому, що маршрутизатори для ухвалення рішення про просування пакета звертаються до адресних таблиць, можна вбачати їхню деяку подібність до мостів і комутаторів. Але природа використовуваних ними адресних таблиць дуже розрізняється. Замість MAC-адрес у таблицях маршрутизації вказуються номери мереж, які поєднуються в інтермережу. Другою відмінністю таблиць маршрутизації від адресних таблиць мостів є спосіб їхнього створення. У той час як міст будує таблицю, пасивно спостерігаючи за інформаційними кадрами, що проходять через нього, і які посилаються кінцевими вузлами мережі один одному, маршрутизатори зі своєї ініціативи обмінюються

спеціальними службовими пакетами, повідомляючи сусідам про відомі їм мережі в інтермережі, маршрутизатори і про зв'язки цих мереж з маршрутизаторами. Зазвичай враховується не тільки топологія зв'язків, але й їхня пропускна здатність і стан. Це дозволяє маршрутизаторам швидше адаптуватися до змін конфігурації мережі, а також правильно передавати пакети в мережах з довільною топологією, яка допускає наявність замкнених контурів.

За допомогою протоколів маршрутизації маршрутизатори складають карту зв'язків мережі того чи іншого ступеня детальності. На підставі цієї інформації для кожного номера мережі приймається рішення про те, якому наступному маршрутизатору слід передавати пакети, які направляються в цю мережу, щоб маршрут виявився раціональним. Результати цих рішень заносяться у таблицю маршрутизації. При зміні конфігурації мережі деякі записи в таблиці стають недійсними. У таких випадках пакети, відправлені по помилкових маршрутах, можуть зациклюватися й губитися. Від того, як швидко протокол маршрутизації приводить у відповідність вміст таблиці реальному стану мережі, залежить якість роботи всієї мережі.

Протоколи маршрутизації можуть бути побудовані на основі різних алгоритмів, які відрізняються способами побудови таблиць маршрутизації, способами вибору найкращого маршруту й інших особливостей своєї роботи.

У всіх описаних вище прикладах під час вибору раціонального маршруту визначався лише наступний (найближчий) маршрутизатор, а не вся послідовність маршрутизаторів від початкового до кінцевого вузла. Відповідно до цього підходу маршрутизація виконується за розподіленою схемою – кожний маршрутизатор відповідальний за вибір тільки одного кроку маршруту, а остаточний маршрут складається в результаті роботи всіх маршрутизаторів, через які проходить даний пакет. Такі алгоритми маршрутизації називаються *однокроковими*.

Існує й протилежний, багатокроковий підхід – *маршрутизація від джерела (Source Routing)*. Відповідно до нього вузол-джерело задає в пакеті, який відправляється в мережу повний маршрут його прямування через всі проміжні маршрутизатори. При використанні багатокрокової маршрутизації немає необхідності будувати й аналізувати таблиці маршрутизації. Це прискорює проходження пакета по мережі, розвантажує маршрутизатори, але при цьому велике навантаження лягає на кінцеві вузли. Ця схема в обчислювальних мережах застосовується сьогодні набагато рідше, ніж схема розподіленої однокрокової маршрутизації. Однак у новій версії протоколу IP

поряд з класичною однокроковою маршрутизацією буде дозволена й маршрутизація від джерела.

Однокрокові алгоритми залежно від способу формування таблиць маршрутизації поділяються на три класи:

- алгоритми фіксованої (статичної) маршрутизації;
- алгоритми простої маршрутизації;
- алгоритми адаптивної (динамічної) маршрутизації.

В алгоритмах *фіксованої маршрутизації* всі записи в таблиці маршрутизації статичні. Адміністратор мережі сам вирішує, на які маршрутизатори слід передавати пакети з тими чи іншими адресами, і вручну (наприклад, за допомогою утиліти route ОС Unix або Windows NT) заносить відповідні записи в таблицю маршрутизації. Таблиця, як правило, створюється в процесі завантаження, надалі вона використовується без змін, доки її вміст не буде відредаговано вручну. Такі виправлення можуть знадобитися, наприклад, якщо в мережі відмовляє який-небудь маршрутизатор і його функції перекладають на інший маршрутизатор. Розрізняють одномаршрутні таблиці, у яких для кожного адресата заданий один шлях, і багатомаршрутні таблиці, що визначають кілька альтернативних шляхів для кожного адресата. У багатомаршрутних таблицях має бути задане правило вибору одного з маршрутів. Найчастіше один шлях є основним, а інші – резервними. Зрозуміло, що алгоритм фіксованої маршрутизації з його ручним способом формування таблиць маршрутизації прийнятний тільки в невеликих мережах з простою топологією. Однак цей алгоритм може бути ефективно використаний і для роботи на магістралях великих мереж, оскільки сама магістраль може мати просту структуру з очевидними найкращими шляхами проходження пакетів у підмережі, приєднані до магістралі.

В алгоритмах *простої маршрутизації* таблиця маршрутизації або зовсім не використовується, або будується без участі протоколів маршрутизації. Виділяють три типи простої маршрутизації:

- *випадкова маршрутизація*, коли прибулий пакет посилається в першому ліпшому напрямі, крім вихідного;
- *лавинна маршрутизація*, коли пакет широкомовно надсилається по всіх можливих напрямках, крім вихідного (аналогічно обробці мостами кадрів з невідомою адресою);
- *маршрутизація за попереднім досвідом*, коли вибір маршруту здійснюється за таблицею, але таблиця будується за принципом моста шляхом аналізу адресних полів пакетів, що з'являються на вхідних портах.

Найпоширенішими є алгоритми *адаптивної (динамічної) маршрутизації*. Ці алгоритми забезпечують автоматичне відновлення таблиць маршрутизації після зміни конфігурації мережі. Протоколи, побудовані на основі адаптивних алгоритмів, дозволяють всім маршрутизаторам збирати інформацію про топологію зв'язків у мережі, оперативно відпрацьовуючи всі зміни конфігурації зв'язків. У таблицях маршрутизації при адаптивній маршрутизації зазвичай є інформація про інтервал часу, протягом якого даний маршрут залишатиметься дійсним. Цей час називають *часом життя маршруту (Time To Live, TTL)*.

Адаптивні алгоритми зазвичай мають розподілений характер, що виражається у відсутні в мережі будь-яких виділених маршрутизаторів, які б збирали і узагальнювали топологічну інформацію. Ця робота розподілена між всіма маршрутизаторами.

Останнім часом позначилася тенденція використання так званих серверів маршрутів. Сервер маршрутів збирає маршрутну інформацію, а потім роздає її за запитами маршрутизаторам, які в цьому випадку звільняються від функції створення таблиць маршрутизації, або створюють тільки частини цих таблиць. З'явилися спеціальні протоколи взаємодії маршрутизаторів із серверами маршрутів, наприклад Next Hop Resolution Protocol (NHRP).

Адаптивні алгоритми маршрутизації мають відповідати декільком важливим вимогам. *По-перше*, вони мають забезпечувати, якщо не оптимальність, то хоча б раціональність маршруту. *По-друге*, алгоритми мають бути досить простими, щоб для їхньої реалізації не витрачалось занадто багато мережних ресурсів, зокрема вони не мають потребувати занадто великого обсягу обчислень або породжувати інтенсивний службовий трафік. *І, нарешті*, алгоритми маршрутизації повинні мати властивість збіжності, тобто завжди приводити до однозначного результату за прийнятний час.

Адаптивні протоколи обміну маршрутною інформацією, що застосовуються на сьогодні в обчислювальних мережах, у свою чергу, поділяються на дві групи, кожна з яких пов'язана з одним з типів алгоритмів:

- дистанційно-векторні алгоритми (Distance Vector Algorithms, DVA);
- алгоритми стану зв'язків (Link State Algorithms, LSA).

В алгоритмах *дистанційно-векторного типу* кожний маршрутизатор періодично й ширококомовно розсилає по мережі вектор, компонентами якого є відстані від даного маршрутизатора до всіх відомих йому мереж. Під відстанню зазвичай розуміють кількість хопів. Можлива й інша метрика, яка враховує не лише кількість проміжних маршрутизаторів, але й час проходження пакетів по мережі між сусідніми маршрутизаторами. Після отримання вектора від

сусіда маршрутизатор нарощує відстані до зазначених у векторі мереж на відстань до даного сусіда. Отримавши вектор від сусіднього маршрутизатора, кожний маршрутизатор додає до нього інформацію про відомі йому інші мережі, про які він довідався безпосередньо (якщо вони підключені до його портів) або з аналогічних оголошень інших маршрутизаторів, а потім знову розсилає нове значення вектора по мережі. Зрештою, кожний маршрутизатор отримує інформацію про всі наявні в інтермережі мережі і про відстань до них через сусідні маршрутизатори.

Дистанційно-векторні алгоритми добре працюють лише в невеликих мережах. У великій мережах вони засмічують лінії зв'язку інтенсивним широкомовним трафіком, до того ж зміни конфігурації можуть відпрацьовуватися за цим алгоритмом не завжди коректно, оскільки маршрутизатори не мають точної уяви про топологію зв'язків у мережі, а мають у своєму розпорядженні лише узагальнену інформацію – вектор дистанцій, до того ж, отриманої через посередників. Робота маршрутизатора відповідно до дистанційно-векторного протоколу нагадує роботу мосту, оскільки точної топологічної картини мережі такий маршрутизатор не має.

Найпоширенішим протоколом, заснованим на дистанційно-векторному алгоритмі, є протокол RIP, розповсюджений у двох версіях – RIP IP, що працює з протоколом IP, і RIP IPX, який працює з протоколом IPX.

Алгоритми стану зв'язків забезпечують кожний маршрутизатор інформацією, достатньою для побудови точного графа зв'язків мережі. Усі маршрутизатори працюють на підставі однакових графів, що робить процес маршрутизації більш стійким до змін конфігурації. «Широкомовне» розсилання (тобто передача пакета всім безпосереднім сусідам маршрутизатора) використовується тут тільки при змінах стану зв'язків, що відбувається в надійних мережах не так часто. Вершинами графа є як маршрутизатори, так і поєднувані ними мережі. Розповсюджувана по мережі інформація складається з опису зв'язків різних типів: маршрутизатор – маршрутизатор, маршрутизатор – мережа.

Щоб зрозуміти, у якому стані перебувають лінії зв'язку, підключені до його портів, маршрутизатор періодично обмінюється короткими пакетами HELLO зі своїми найближчими сусідами. Цей службовий трафік також засмічує мережу, але не в такому ступені як, наприклад, RIP-пакети, оскільки пакети HELLO мають набагато менший обсяг.

Протоколами, заснованими на алгоритмі стану зв'язків, є протоколи IS-IS (Intermediate System to Intermediate System) стека OSI, OSPF (Open

Shortest Path First) стека TCP/IP і нещодавно реалізований протокол NLSP стека Novell.

12.2 Функції маршрутизатора

Основна функція маршрутизатора – читання заголовків пакетів мережних протоколів, прийнятих і буферизованих по кожному порту (наприклад, IPX, IP, AppleTalk або DECnet), і ухвалення рішення про подальший маршрут проходження пакета по його мережній адресі, який включає, як правило, номер мережі й номер вузла.

Функції маршрутизатора можуть бути розбиті на 3 групи відповідно до рівнів моделі OSI (рис. 12.1).



Рисунок 12.1 – Функціональна модель маршрутизатора

Рівень інтерфейсів

На нижньому рівні маршрутизатор, як і будь-який пристрій, підключений до мережі, забезпечує фізичний інтерфейс з середовищем передачі, включаючи узгодження рівнів електричних сигналів, лінійне й логічне кодування, оснащення певним типом розніму. У різних моделях маршрутизаторів часто передбачаються різні набори фізичних інтерфейсів, які є комбінацією портів для приєднання локальних і глобальних мереж. З кожним інтерфейсом для підключення локальної мережі нерозривно пов'язаний певний протокол канального рівня, наприклад, Ethernet, Token Ring, FDDI. Інтерфейси для приєднання до глобальних мереж найчастіше визначають тільки деякий стандарт фізичного рівня, над яким у маршрутизаторі можуть працювати різні протоколи канального рівня. Наприклад, глобальний порт може підтримувати інтерфейс V.35, над яким можуть працювати протоколи канального рівня: LAP-B (що використовується у мережах X.25), LAP-F (який використовується у мережах frame relay), LAP-D (використовуваний у мережах ISDN). Різниця між інтерфейсами локальних і глобальних мереж пояснюється тим, що технології локальних мереж працюють за власними стандартами фізичного рівня, які, як правило, не можуть використовуватися в інших технологіях, тому інтерфейс для локальної мережі є сполученням фізичного й канального рівнів і називається по імені відповідної технології – наприклад, інтерфейс Ethernet.

Інтерфейси маршрутизатора виконують повний набір функцій фізичного й канального рівнів по передачі кадра, включаючи отримання доступу до середовища (якщо це необхідно), формування бітових сигналів, прийом кадра, підрахунок його контрольної суми й передачу поля даних кадра верхньому рівню у випадку, якщо контрольна сума має коректне значення.

Як і будь-який кінцевий вузол, кожен порт маршрутизатора має власну апаратну адресу (у локальних мережах MAC-адреса), по якій іншими вузлами мережі йому й направляються кадри, що вимагають маршрутизації.

Перелік фізичних інтерфейсів, які підтримує та або інша модель маршрутизатора, є його найважливішою споживчою характеристикою. Маршрутизатор має підтримувати всі протоколи канального й фізичного рівнів, що використовуються в кожній з мереж, до яких він буде безпосередньо приєднаний. На рис. 12.1 показана функціональна модель маршрутизатора з чотирма портами, які реалізують такі фізичні інтерфейси: 10Base-T і 10Base-2 для двох портів Ethernet, UTP для Token Ring і V.35, над яким можуть працювати протоколи LAP-B, LAP-D або LAP-F, забезпечуючи підключення до мереж X.25, ISDN або frame relay.

Кадри, які надходять на порти маршрутизатора, після обробки відповідними протоколами фізичного й канального рівнів, звільняються від заголовків канального рівня. Пакети, витягнуті з поля даних кадра, передаються модулю мережного протоколу.

Рівень мережного протоколу

Мережний протокол у свою чергу витягає з пакета заголовки мережного рівня й аналізує вміст його полів. Насамперед перевіряється контрольна сума, і якщо пакет прийшов ушкодженим, то він відкидається. Виконується перевірка, чи не перевищив час, який провів пакет у мережі (час життя пакета), припустимої величини. Якщо перевищив, то пакет також відкидається. На цьому етапі вносяться коректування у вміст деяких полів, наприклад, нарощується час життя пакета, перераховується контрольна сума.

На мережному рівні виконується одна з найважливіших функцій маршрутизатора – фільтрація трафіка. Маршрутизатор, маючи вищий інтелект, ніж мости й комутатори, дозволяє задавати й може відпрацьовувати значно складніші правила фільтрації. Пакет мережного рівня, який перебуває в полі даних кадра, для мостів/комутаторів видається неструктурованою двійковою послідовністю. Маршрутизатори ж, програмне забезпечення яких містить модуль мережного протоколу, здатні робити розбирання і аналіз окремих полів пакета. Вони оснащуються розвинутими засобами користувальницького інтерфейсу, які дозволяють адміністратору без особливих зусиль задавати складні правила фільтрації. Вони, наприклад, можуть заборонити проходження в корпоративну мережу всіх пакетів, окрім пакетів, які надходять з підмереж цього самого підприємства. Фільтрація в даному випадку проводиться по мережних адресах, і всі пакети, адреси яких не входять у дозволений діапазон, відкидаються. Маршрутизатори, як правило, також можуть аналізувати структуру повідомлень транспортного рівня, тому фільтри можуть не пропускати в мережу повідомлення певних прикладних служб, наприклад, служби tehet, аналізуючи поле типу протоколу у транспортному повідомленні.

У випадку, якщо інтенсивність надходження пакетів вище інтенсивності, з якою вони обробляються, пакети можуть утворити чергу. Програмне забезпечення маршрутизатора може реалізувати різні дисципліни обслуговування черг пакетів: у порядку надходження за принципом «перший прийшов – першим обслужений» (First Input First Output, FIFO), випадкове раннє виявлення, коли обслуговування йде за правилом FIFO, але при досягненні довжиною черги деякого граничного значення нові пакети, що

надходять, відкидаються (Random Early Detection, RED), а також різні варіанти пріоритетного обслуговування.

До мережного рівня належить основна функція маршрутизатора – визначення маршруту пакета. За номером мережі, витягнутим із заголовка пакета, модуль мережного протоколу знаходить у таблиці маршрутизації рядок, який містить мережну адресу наступного маршрутизатора, і номер порту, на який потрібно передати даний пакет, щоб він рухався у правильному напрямі. Якщо в таблиці відсутній запис про мережу призначення пакета й до того ж немає запису про маршрутизатор за замовчуванням, то даний пакет відкидається.

Перед тим як передати мережну адресу наступного маршрутизатора на канальний рівень, необхідно перетворити її в локальну адресу тієї технології, що використовується в мережі, яка містить наступний маршрутизатор. Для цього мережний протокол звертається до *протоколу дозволу адрес*. Протоколи цього типу встановлюють відповідність між мережними й локальними адресами або на підставі заздалегідь складених таблиць, або шляхом розсилання широкомовних запитів. Таблиця відповідності локальних адрес мережним адресам будується окремо для кожного мережного інтерфейсу. Протоколи дозволу адрес займають проміжне положення між мережним і канальним рівнями.

З мережного рівня пакет, локальна адреса наступного маршрутизатора й номер порту маршрутизатора передаються вниз, канальному рівню. На підставі зазначеного номера порту здійснюється комутація з одним із інтерфейсів маршрутизатора, засобами якого виконується впакування пакета у кадр відповідного формату. У полі адреси призначення заголовка кадра міститься локальна адреса наступного маршрутизатора. Готовий кадр відправляється в мережу.

Рівень протоколів маршрутизації

Мережні протоколи активно використовують у своїй роботі таблицю маршрутизації, але ні її побудовою, ні підтримкою її вмісту не займаються. Ці функції виконують протоколи маршрутизації. На підставі цих протоколів маршрутизатори обмінюються інформацією про топологію мережі, а потім аналізують отримані відомості, визначаючи найкращі за тими чи іншими критеріями маршрути. Результати аналізу і являють вміст таблиць маршрутизації.

Крім перерахованих вище функцій, на маршрутизатори можуть покладатися й інші обов'язки, наприклад, операції, пов'язані з фрагментацією.

Детальніше робота маршрутизаторів буде описана в ході розгляду конкретних протоколів мережного рівня.

12.3 Реалізація міжмережної взаємодії засобами TCP/IP

На сьогодні стек TCP/IP є найпопулярнішим засобом організації складених мереж. До 1996 року безперечним лідером був стек IPX/SPX компанії Novell, але потім картина різко змінилася – стек TCP/IP за темпами зростання кількості установок став набагато випереджати інші стеки, а з 1998 року вийшов у лідери й в абсолютному вираженні. Саме тому подальше вивчення функцій мережного рівня проводитиметься на прикладі стека TCP/IP.

Багаторівнева структура стека TCP/IP

У стеку TCP/IP визначено 4 рівні (рис. 12.2). Кожен із цих рівнів несе деяке навантаження за рішенням основної задачі – організації надійної й продуктивної роботи складеної мережі, частини якого побудовані на основі різних мережних технологій.

Рівень I	Прикладний рівень
Рівень II	Основний (транспортний) рівень
Рівень III	Рівень міжмережної взаємодії
Рівень IV	Рівень мережних інтерфейсів

Рисунок 12.2 – Багаторівнева архітектура стека TCP/IP

Рівень міжмережної взаємодії

Стрижнем всієї архітектури є *рівень міжмережної взаємодії*, що реалізує концепцію передачі пакетів у режимі без встановлення з'єднань, тобто дейтаграмним способом. Саме цей рівень забезпечує можливість переміщення пакетів по мережі, використовуючи той маршрут, який у цей момент є найраціональнішим. Цей рівень також називають рівнем internet, вказуючи тим самим на основну його функцію – передачу даних через складену мережу.

Основним протоколом мережного рівня (у термінах моделі OSI) у стеку є протокол IP (Internet Protocol). Цей протокол від самого початку проектувався як протокол передачі пакетів у складених мережах, що складаються з великої

кількості локальних мереж, об'єднаних як локальними, так і глобальними зв'язками. Тому протокол IP добре працює в мережах зі складною топологією, раціонально використовуючи наявність у них підсистем і ощадливо витрачаючи пропускну здатність низькошвидкісних ліній зв'язку. Оскільки протокол IP є дейтаграмним протоколом, він не гарантує доставку пакетів до вузла призначення, але намагається це зробити.

До рівня міжмережної взаємодії також належать усі протоколи, пов'язані зі складанням і модифікацією таблиць маршрутизації, такі як протоколи збирання маршрутної інформації RIP (Routing Internet Protocol) і OSPF (Open Shortest Path First), а також протокол міжмережних управляючих повідомлень ICMP (Internet Control Message Protocol). Останній протокол призначений для обміну інформацією про помилки між маршрутизаторами мережі й вузлом-джерелом пакета. За допомогою спеціальних пакетів ICMP повідомляє про неможливість доставки пакета, про перевищення часу життя або тривалості зборки пакета із фрагментів, про аномальні величини параметрів, про зміну маршруту пересилання й типу обслуговування, про стан системи тощо.

12.4 Контрольні запитання та завдання

1. Дайте відповідь, у чому проявляється ненадійність протоколу IP?
2. Чи можуть бути виявлені помилки на рівні Internet? Чи можуть вони бути виправлені засобами цього рівня?
3. Скільки рівнів має стек протоколів TCP/IP? Які їхні функції? Які особливості цього стека обумовлюють його лідируюче положення у світі мережних технологій?
4. Порівняйте функції маршрутизаторів, які підтримують маршрутизацію від джерела, з функціями маршрутизаторів, що підтримують протоколи адаптивної маршрутизації.
5. Які метрики відстані можуть бути використані в алгоритмах збору маршрутної інформації?
6. Порівняйте інтенсивність широкомовного трафіка, який породжено протоколами RIP і OSPF.
7. Визначте, які елементи мережі можуть виконувати фрагментацію:
 - тільки комп'ютери;
 - тільки маршрутизатори;
 - комп'ютери, маршрутизатори, мости, комутатори;
 - комп'ютери й маршрутизатори?

8. Дайте відповідь, що відбудеться, якщо при передачі пакета він був фрагментований і один із фрагментів не дійшов до вузла призначення після закінчення тайм-ауту:

- модуль IP вузла-відправника повторить передачу фрагмента, який не дійшов;

- модуль IP вузла-відправника повторить передачу всього пакета, до складу якого входив фрагмента, який не дійшов;

- модуль IP вузла-отримувача відкине всі отримані фрагменти пакета, у якому загубився один фрагмент;

- модуль IP вузла-відправника не застосовуватиме ніяких дій щодо повторної передачі даного пакета?

Лекція 13

ПРИНЦИПИ ПОБУДОВИ СТЕКА TCP/IP

13.1 Функції стеку TCP/IP

Оскільки на мережному рівні не встановлюються з'єднання, то немає ніяких гарантій, що всі пакети будуть доставлені в місце призначення цілими і непошкодженими або прийдуть у такому самому порядку, у якому вони були відправлені. Цю задачу – забезпечення надійного інформаційного зв'язку між двома кінцевими вузлами – вирішує *основний рівень* стеку TCP/IP, названий також *транспортним*.

На цьому рівні функціонують протокол управління передачею TCP (Transmission Control Protocol) і протокол дейтаграм користувача UDP (User Datagram Protocol). Протокол TCP забезпечує надійну передачу повідомлень між віддаленими прикладними процесами за рахунок створення логічних з'єднань. Цей протокол дозволяє рівноранговим об'єктам на комп'ютері-відправнику й комп'ютері-отримувачі підтримувати обмін даними в дуплексному режимі. TCP дозволяє без помилок доставити сформований на одному з комп'ютерів потік байт у будь-який інший комп'ютер, який входить у складену мережу. TCP розділяє потік байт на частини – *сегменти*, і передає їх рівню міжмережної взаємодії, що лежить нижче. Після того як ці сегменти будуть доставлені засобами міжмережного рівня взаємодії в пункт призначення, протокол TCP знову збере їх у безперервний потік байт.

Протокол UDP забезпечує передачу прикладних пакетів дейтаграмним способом, як і головний протокол міжмережного рівня взаємодії IP, і виконує тільки функції сполучної ланки (мультиплексора) між мережним протоколом і численними службами прикладного рівня або користувальницьких процесів.

Прикладний рівень

Прикладний рівень поєднує всі служби, що надаються системою користувальницьким додаткам. За довгі роки використання в мережах різних країн і організацій стек TCP/IP надбав велику кількість протоколів і служб прикладного рівня. Прикладний рівень реалізується програмними системами, побудованими в архітектурі клієнт-сервер, що базуються на протоколах нижніх рівнів. На відміну від протоколів інших трьох рівнів, протоколи прикладного

рівня займаються деталями конкретного додатка й «не цікавляться» способами передачі даних по мережі. Цей рівень постійно розширюється за рахунок приєднання до старих, які пройшли багаторічну експлуатацію, мережних служб типу Telnet, FTP, TFTP, DNS, SNMP порівняно нових служб таких, наприклад, як протокол передачі гіпертекстової інформації HTTP.

Рівень мережних інтерфейсів

Ідеологічною відмінністю архітектури стека TCP/IP від багаторівневої організації інших стеків є інтерпретація функцій самого нижнього рівня – *рівня мережних інтерфейсів*. Протоколи цього рівня мають забезпечувати інтеграцію в складену мережу інших мереж, причому задача ставиться так: мережа TCP/IP повинна мати засоби приєднання до себе будь-якої іншої мережі, яку б внутрішню технологію передачі даних ця мережа не використовувала. Звідси випливає, що цей рівень не можна означити раз і назавжди. Для кожної технології, що входить в складену мережу підмережі, мають бути розроблені власні інтерфейсні засоби. До таких інтерфейсних засобів належать протоколи інкапсуляції IP-пакетів рівня міжмережної взаємодії в кадри локальних технологій. Наприклад, документ RFC 1042 визначає способи інкапсуляції IP-пакетів у кадри технологій IEEE 802. Для цих цілей має використовуватися заголовок LLC/SNAP, причому в полі Type заголовка SNAP має бути зазначений код 0x0800. Тільки для протоколу Ethernet в RFC 1042 зроблено виняток – крім заголовка LLC/ SNAP дозволяється використовувати кадр Ethernet DIX, який не має заголовка LLC, зате має поле Type. У мережах Ethernet кращим є інкапсуляція IP-пакета в кадр Ethernet DIX.

Рівень мережних інтерфейсів у протоколах TCP/IP не регламентується, але він підтримує всі популярні стандарти фізичного й канального рівнів: для локальних мереж це Ethernet, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, 100VG-AnyLAN, для глобальних – протоколи з'єднань «точка-точка» SLIP і PPP, протоколи територіальних мереж з комутацією пакетів X.25, frame relay. Розроблена також спеціальна специфікація, яка визначає використання технології ATM як транспорт канального рівня. Зазвичай з появою нової технології локальних або глобальних мереж вона швидко включається в стек TCP/IP за рахунок розробки відповідного RFC, що визначає метод інкапсуляції IP-пакетів у її кадри (специфікація RFC 1577, яка визначає роботу IP через мережі ATM, з'явилася в 1994 році незабаром після прийняття основних стандартів цієї технології).

Відповідність рівнів стека TCP/IP семирівневій моделі ISO/OSI

Оскільки стік TCP/IP був розроблений до появи моделі взаємодії відкритих систем ISO/OSI, то, хоча він також має багаторівневу структуру, відповідність рівнів стека TCP/IP рівням моделі OSI досить умовна (рис. 13.1). Розглядаючи багаторівневу архітектуру TCP/IP, можна виділити в ній, подібно архітектурі OSI, рівні, функції яких залежать від конкретної технічної реалізації мережі, і рівні, функції яких орієнтовані на роботу з додатками (рис. 13.2).

7	WWW, Gopher, WAIS	SNMP	FTP	telnet	SMTP	TFTP	I
6							
5	TCP					UDP	II
4							
3	IP	ICMP	RIP	OSPF	ARP		III
2	Не регламентується Ethernet, Token Ring, FDDI, X.25, SLIP, PPP						IV
1							

Рівні моделі OSI

Рівні стека TCP/IP

Рисунок 13.1 – Відповідність рівнів стека TCP/IP семирівневій моделі OSI

Протоколи прикладного рівня стека TCP/IP працюють на комп'ютерах, які виконують додатки користувачів. Навіть повна зміна мережного обладнання в загальному випадку не має впливати на роботу додатків, якщо вони отримують доступ до мережних можливостей через протоколи прикладного рівня.

Протоколи транспортного рівня вже більше залежать від мережі, оскільки вони реалізують інтерфейс до рівнів, які безпосередньо організують передачу даних по мережі. Однак, подібно протоколам прикладного рівня, програмні модулі, які реалізують протоколи транспортного рівня, встановлюються лише на кінцевих вузлах. Протоколи двох нижніх рівнів мережозалежні, а отже, програмні модулі протоколів міжмережного рівня й рівня мережних інтерфейсів встановлюються як на кінцевих вузлах складеної мережі, так і на маршрутизаторах.

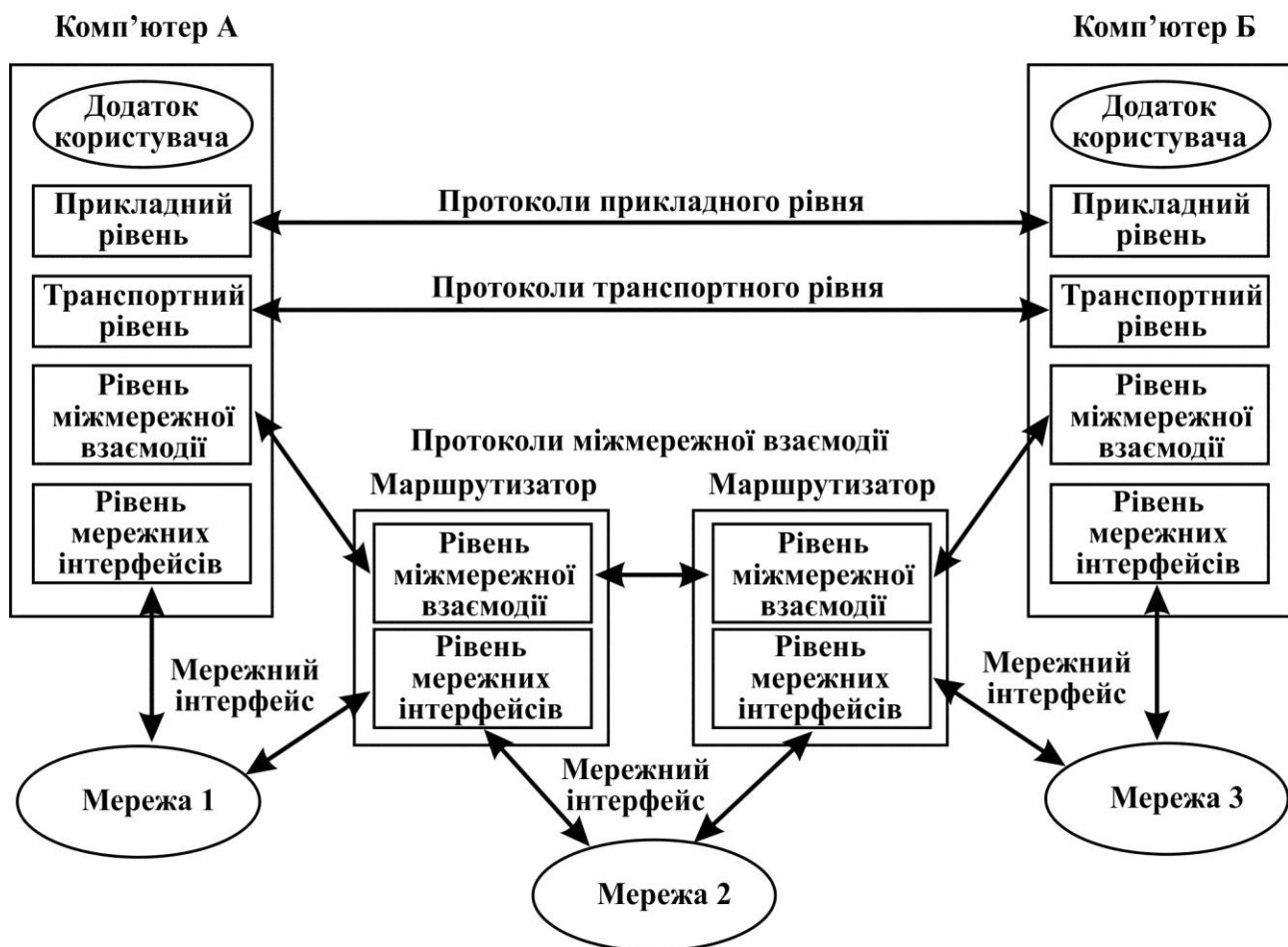


Рисунок 13.2 – Мережезалежні та мереженезалежні рівні стека TCP/IP

Кожний комунікаційний протокол оперує з деякою одиницею переданих даних. Назви цих одиниць іноді закріплюються стандартом, а частіше просто визначаються традицією. У стеку TCP/IP за багато років його існування утворилася усталена термінологія в цій галузі (рис. 13.3).

Потоком називають дані, що надходять від додатків на вхід протоколів транспортного рівня TCP і UDP.

Протокол TCP нарізає з потоку даних *сегменти*.

Одиницю даних протоколу UDP часто називають *дейтаграмою* (або датаграмою). Дейтаграма – це загальна назва для одиниць даних, якими оперують протоколи без установлення з'єднань. До таких протоколів належить і протокол міжмережної взаємодії IP.

Дейтаграму протоколу IP ще називають *пакетом*.

У стеку TCP/IP прийнято називати *кадрами* (*фреймами*) одиниці даних протоколів, на основі яких IP-пакети переносяться через підмережі складеної мережі. При цьому не має значення, яке назва використовується для цієї одиниці даних у локальній технології.

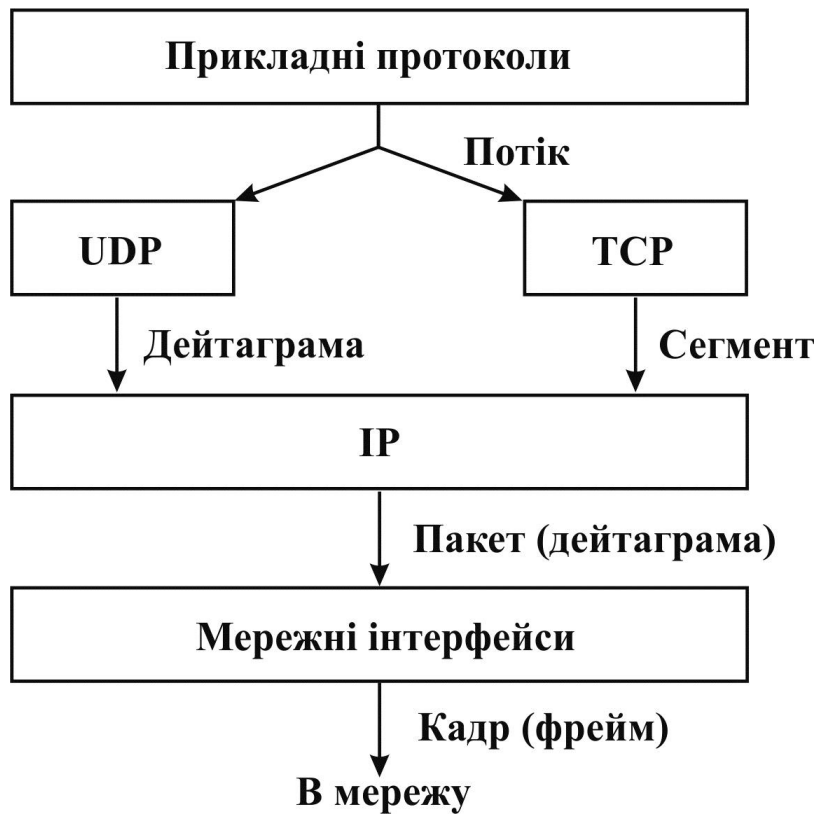


Рисунок 13.3 – Назва одиниць даних, які використовуються в TCP/IP

Висновки до підрозділу

- Складена мережа (internetwork або internet) – це сукупність декількох мереж, які також називаються підмережами (subnet), що з'єднуються між собою маршрутизаторами. Організація спільної транспортної служби в складеній мережі називається міжмережною взаємодією (internetworking).
- До функції мережного рівня належать: передача пакетів між кінцевими вузлами в складених мережах, вибір маршруту, узгодження локальних технологій окремих підмереж.
- Маршрут – це послідовність маршрутизаторів, які має пройти пакет від відправника до пункту призначення. Задачу вибору маршруту з декількох можливих вирішують маршрутизатори й кінцеві вузли на основі таблиць маршрутизації. Записи в таблицю можуть заносятися вручну адміністратором і автоматично протоколами маршрутизації.
- Протоколи маршрутизації (наприклад, RIP або OSPF) слід відрізняти від власне мережних протоколів (наприклад, IP або IPX). У той час як перші збирають і передають по мережі виключно службову інформацію про можливі маршрути, другі призначені для передачі користувальницьких даних.

- Мережні протоколи й протоколи маршрутизації реалізуються у вигляді програмних модулів на кінцевих вузлах-комп'ютерах і на проміжних вузлах-маршрутизаторах.

- Маршрутизатор є складним багатофункціональним пристроєм, задачами якого є: побудова таблиці маршрутизації, визначення на її основі маршруту, буферизація, фрагментація та фільтрація пакетів, що надходять, підтримка мережних інтерфейсів. Функції маршрутизаторів можуть виконувати як спеціалізовані пристрої, так і універсальні комп'ютери з відповідним програмним забезпеченням.

- Для алгоритмів маршрутизації характерні однокроковий та багатокроковий підходи. Однокрокові алгоритми поділяються на алгоритми фіксованої, простої й адаптивної маршрутизації. Адаптивні протоколи маршрутизації є найпоширенішими й у свою чергу можуть ґрунтуватися на дистанційно-векторних алгоритмах і алгоритмах стану зв'язків.

- Найбільшого поширення для побудови складених мереж останнім часом отримав стек TCP/IP. Стек TCP/IP має 4 рівні: прикладний, основний, міжмережної взаємодії та мережних інтерфейсів. Відповідність рівнів стека TCP/IP рівням моделі OSI досить умовна.

- *Прикладний рівень* поєднує всі служби, що надаються системою користувальницьким додаткам: традиційні мережні служби типу telnet, FTP, TFTP, DNS, SNMP, а також порівняно нові, такі, як протокол передачі гіпертекстової інформації HTTP.

- *На основному рівні* стека TCP/IP, який ще називається транспортним, функціонують протоколи TCP і UDP. Протокол управління передачею TCP вирішує задачу забезпечення надійного інформаційного зв'язку між двома кінцевими вузлами. Дейтаграмний протокол UDP використовується як економічний засіб зв'язку рівня міжмережної взаємодії з прикладним рівнем.

- *Рівень міжмережної взаємодії* реалізує концепцію комутації пакетів у режимі без встановлення з'єднань. Основними протоколами цього рівня є дейтаграмний протокол IP і протоколи маршрутизації (RIP, OSPF, BGP тощо). Допоміжну роль виконують протокол міжмережних управляючих повідомлень ICMP, протокол групового управління IGMP і протокол дозволу адрес ARP.

- Протоколи *рівня мережних інтерфейсів* забезпечують інтеграцію в складену мережу інших мереж. Цей рівень не регламентується, але підтримує всі популярні стандарти фізичного й канального рівнів: для локальних мереж – Ethernet, Token Ring, FDDI та інші, для глобальних мереж – X.25, frame relay, PPP, ISDN тощо.

- У стеку TCP/IP для іменування одиниць переданих даних на різних рівнях використовують різні назви: потік, сегмент, дейтаграма, пакет, кадр.

13.2 Організація доменів та доменних імен

Для ідентифікації комп'ютерів апаратне й програмне забезпечення в мережах TCP/IP покладається на IP-адреси, тому для доступу до мережного ресурсу в параметрах програми цілком достатньо вказати IP-адреси, щоб програма правильно зрозуміла, до якого хосту їй потрібно звернутися. Наприклад, команда `ftp://192.45.66.17` встановлюватиме сеанс зв'язку з потрібним ftp-сервером, а команда `http://203.23.106.33` відкриє початкову сторінку на корпоративному Web-сервері. Однак користувачі зазвичай бажають працювати з символьними іменами комп'ютерів, і операційні системи локальних мереж привчили їх до цього зручного способу. Отже, у мережах TCP/IP мають існувати символьні імена хостів і механізм для встановлення відповідності між символьними іменами й IP-адресами.

В операційних системах, які спочатку розроблялися для роботи в локальних мережах, таких як Novell NetWare, Microsoft Windows або IBM OS/2, користувачі завжди працювали з символьними іменами комп'ютерів. Оскільки локальні мережі склалися з невеликої кількості комп'ютерів, то використовувалися так звані плоскі імена, тобто які складаються з послідовності символів, не розділених на частини. Прикладами таких імен є: `NW1_1`, `mail2`, `MOSCOW_SALES_2`. Для встановлення відповідності між символьними іменами й MAC-адресами в цих операційних системах застосовувався механізм ширококомовних запитів, подібний до механізму запитів протоколу ARP. Так, ширококомовний спосіб дозволу імен реалізований у протоколі NetBIOS, на якому було побудовано багато локальних ОС. Так звані NetBIOS-імена стали на довгі роки одним із основних типів плоских імен у локальних мережах.

Для стека TCP/IP, розрахованого в загальному випадку на роботу в великих територіально розподілених мережах, подібний підхід виявляється неефективним з кількох причин.

Плоскі імена не дають можливості розробити єдиний алгоритм забезпечення унікальності імен у межах великої мережі. У невеликих мережах унікальність імен комп'ютерів забезпечує адміністратор мережі, записуючи кілька десятків імен у журналі або файлі. Зі збільшенням мережі задачу вирішують уже декілька адміністраторів, погоджуючи імена між собою

неформальним способом. Але якщо мережа розташована в різних містах або країнах, то адміністраторам кожної частини мережі слід придумати спосіб іменування, який дозволив би їм давати імена новим комп'ютерам незалежно від інших адміністраторів, в той же час забезпечуючи унікальність імен для всієї мережі. Самий надійний спосіб вирішення цієї задачі – принципова відмова від плоских імен.

Широкомовний спосіб встановлення відповідності між символьними іменами й локальними адресами добре працює лише в невеликій локальній мережі, не розділеній на підмережі. У великих мережах, де загальна широкомовність не підтримується, потрібний інший спосіб дозволу символьних імен. Зазвичай гарною альтернативою широкомовності є застосування централізованої служби, яка підтримує відповідність між різними типами адрес всіх комп'ютерів мережі. Компанія Microsoft для своєї корпоративної операційної системи Windows NT розробила централізовану службу WINS, яка підтримує базу даних NetBIOS-імен і відповідних їм IP-адрес.

Для ефективної організації іменування комп'ютерів у великих мережах природним є застосування ієрархічних складених імен.

У стеку TCP/IP застосовується доменна система імен, що має ієрархічну деревоподібну структуру, яка допускає використання в імені довільної кількості складових частин (рис. 13.4).

Ієрархія доменних імен аналогічна ієрархії імен файлів, прийнятій в багатьох популярних файлових системах. Дерево імен починається з кореня, позначеного тут крапкою (.). Потім іде старша символьна частина імені, друга за старшинством символьна частина імені й т.д. Молодша частина імені відповідає кінцевому вузлу мережі. На відміну від імен файлів, при записі яких спочатку вказується сама старша складова, потім складова більш низького рівня й т.д., запис доменного імені починається з самої молодшої складової, а закінчується самою старшою. Складові частини доменного імені відділяється одна від одної крапкою. Наприклад, в імені `partnering.microsoft.com` складова `partnering` є ім'ям одного з комп'ютерів у домені `Microsoft.com`.

Поділ імені на частини дозволяє *розділити адміністративну відповідальність* за призначення унікальних імен між різними людьми або організаціями в межах свого рівня ієрархії. Так для прикладу, наведеного на рис. 13.4, одна людина може відповідати за те, щоб усі імена, які мають закінчення «та», мали унікальну спадаючу вниз по ієрархії частину. Якщо ця людина справляється зі своїми обов'язками, то всі імена типу `www.ru`, `mail.mmt.ru` або `m2.zil.mmt.ru` відрізнятимуться другою за старшинством частиною.

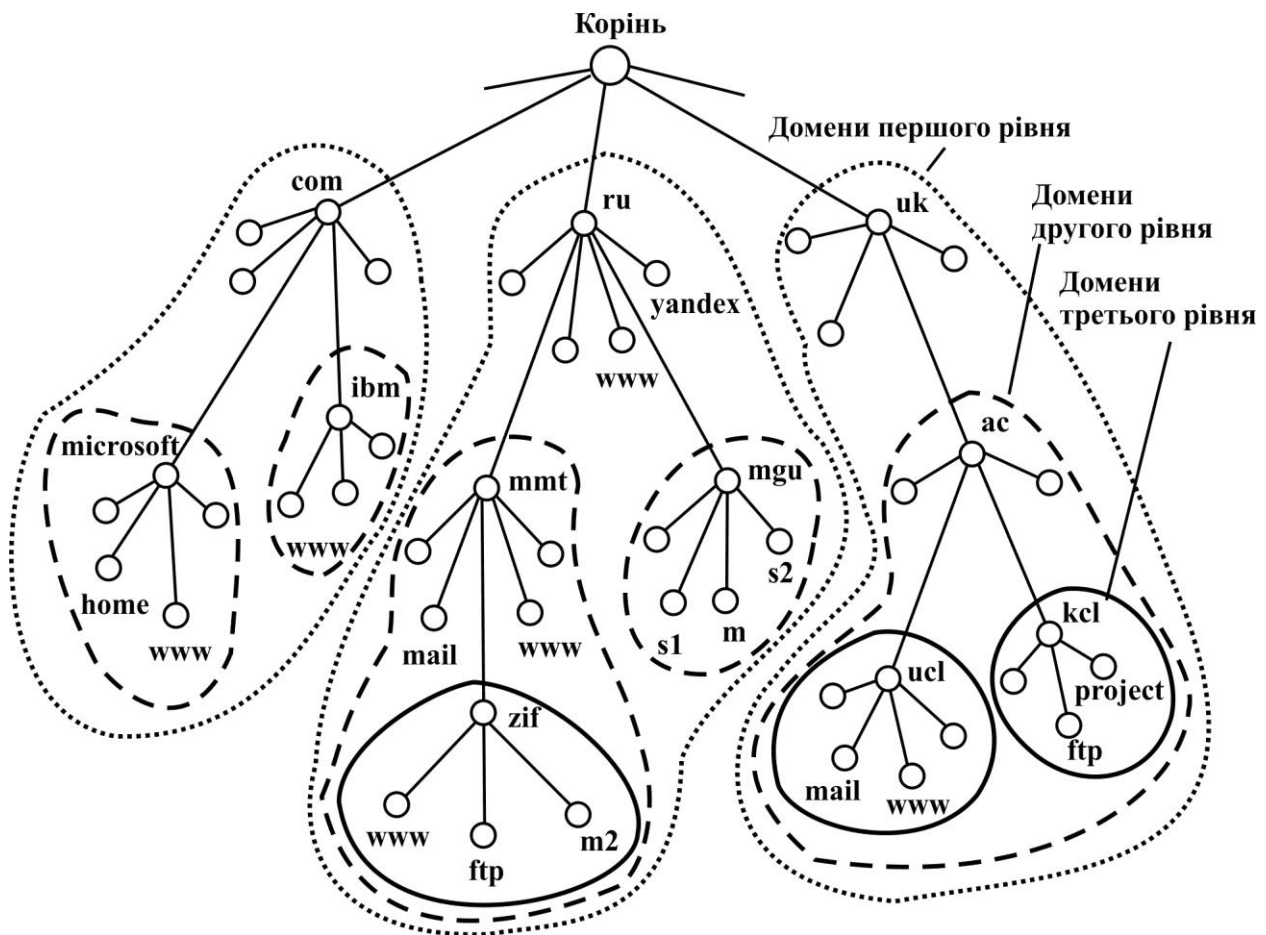


Рисунок 13.4 – Простір доменних імен

Поділ адміністративної відповідальності дозволяє вирішити проблему створення унікальних імен без взаємних консультацій між організаціями, відповідальними за імена одного рівня ієрархії. Очевидно, що має існувати одна організація, відповідальна за призначення імен верхнього рівня ієрархії.

Сукупність імен, у яких кілька старших складових частин збігаються, утворюють *домен* імен (*domain*). Наприклад, імена `www1.zil.mmt.ru`, `ftp.zil.mmt.ru`, `yandex.ru` і `sl.mgu.ru` входять у домен `ru`, оскільки всі ці імена мають одну загальну старшу частину – ім'я `ru`. Іншим прикладом є домен `mgu.ru`. Із представлених на рис. 10.9 імен у нього входять імена `sl.mgu.ru`, `s2.mgu.ru` і `rn.mgu.ru`. Цей домен утворюють імена, в яких дві старші частини завжди рівні `rngu.ru`. Ім'я `www.mmt.ru` у домен `mgu.ru` не входить, оскільки має складову `mmt`, яка відрізняється.

Термін «домен» багатозначний, тому його потрібно трактувати в рамках певного контексту. Крім доменів імен стека TCP/IP також часто згадуються домени Windows NT, домени колізій тощо. Загальним у всіх цих термінів є те, що вони описують деяку множину комп'ютерів, яка має будь-яку певну властивість.

Якщо один домен входить в інший домен як його складова частина, то такий домен можуть називати *піддоменом (subdomain)*, хоча назва домен за ним також залишається. Зазвичай піддомен називають по імені тієї його старшої складової, що відрізняє його від інших піддоменів. Наприклад, піддомен mmt.ru зазвичай називають піддоменом (або діменом) mmt. Ім'я піддомену призначає адміністратор вищого домена. Гарною аналогією домена є каталог файлової системи.

Якщо в кожному домені й піддомені забезпечується унікальність імен наступного рівня ієрархії, то й уся система імен складатиметься з унікальних імен.

За аналогією з файловою системою, у доменній системі імен розрізняють короткі імена, відносні імена й повні доменні імена. Коротке ім'я – це ім'я кінцевого вузла мережі: хоста або порту маршрутизатора. Коротке ім'я – це лист дерева імен. Відносне ім'я – це складене ім'я, яке починається з деякого рівня ієрархії, але не самого верхнього. Наприклад, wwwi.zil – це відносне ім'я. *Повне доменне ім'я (fully qualified domain name, FQJDN)* містить складові всіх рівнів ієрархії, починаючи від короткого імені і закінчуючи кореневою точкою: wwwi.zil.mmt.ru.

Необхідно підкреслити, що комп'ютери входять у домен згідно зі своїми складеними іменами, при цьому вони можуть мати зовсім різні IP-адреси, які належать до різних мереж і підмереж. Наприклад, у домен mgu.ru можуть входити хости з адресами 132.13.34.15, 201.22.100.33, 14.0.0.6. Доменна система імен реалізована в мережі Internet, але вона може працювати і як автономна система імен у великій корпоративній мережі, що використовує стік TCP/IP, але не пов'язана з Internet.

В Internet кореневий домен управляється центром InterNIC. Домени верхнього рівня призначаються для кожної країни, а також на організаційній основі. Імена цих доменів мають дотримуватися міжнародного стандарту ISO 3166. Для позначення країн використовуються трибуквені й двубуквенні аббревіатури, а для різних типів організацій – такі позначення:

- com – комерційні організації (наприклад, microsoft.com);
- edu – освітні (наприклад, mitedu);
- gov – урядові організації (наприклад, nsf.gov);
- org – некомерційні організації (наприклад, fidonet.org);
- net – організації, що підтримують мережі (наприклад, nsf.net).

Кожний домен адмініструється окремою організацією, що звичайно розбиває свій домен на піддомени й передає функції адміністрування цих

піддоменів іншим організаціям. Щоб отримати доменне ім'я, необхідно зареєструватися в будь-якій організації, який InterNIC делегували свої повноваження по розподілу імен доменів.

13.3 Система доменних імен DNS

Відповідність між доменними іменами й IP-адресами може встановлюватися як засобами локального хоста, так і засобами централізованої служби. На ранньому етапі розвитку Internet на кожному хості вручну створювався текстовий файл з відомим ім'ям hosts. Цей файл складався з деякої кількості рядків, кожен з яких містив одну пару «IP-адреса – доменне ім'я», наприклад, 102.54.94.97 – rhino.acme.com. Зі зростанням Internet файли hosts також росли, і створення масштабованого рішення для дозволу імен стало необхідністю.

Таким рішенням стала спеціальна служба – *система доменних імен (Domain Name System, DNS)*. DNS – це централізована служба, заснована на розподіленій базі відображень «доменне ім'я – IP-адреса». Служба DNS використовує у своїй роботі протокол типу «клієнт–сервер». У ньому визначені DNS-сервери й DNS-клієнти. DNS-сервери підтримують розподілену базу відображень, а DNS-клієнти звертаються до серверів з запитами про дозвіл доменного імені в IP-адресу.

Служба DNS використовує текстові файли майже такого формату, як і файл hosts, і ці файли адміністратор також підготує вручну. Однак служба DNS опирається на ієрархію доменів, і кожен сервер служби DNS зберігає тільки частину імен мережі, а не всі імена, як це відбувається під час використання файлів hosts. При зростанні кількості вузлів у мережі проблема масштабування вирішується створенням нових доменів і піддоменів імен і додаванням у службу DNS нових серверів.

Для кожного домена імен створюється свій DNS-сервер. Цей сервер може зберігати відображення «доменне ім'я – IP-адреса» для всього домена, включаючи всі його піддомени. Але при цьому рішення виявляється погано масштабованим, оскільки при додаванні нових піддоменів навантаження на цей сервер може перевищити його можливості. Частіше сервер домена зберігає лише імена, які закінчуються на наступному рівні ієрархії в порівнянні з ім'ям домена. (Аналогічно каталогу файлової системи, що містить запис про файли й підкаталоги, безпосередньо тих, що в нього «входять».) Саме при такій організації служби DNS навантаження з дозволу імен розподіляється

більш-менш рівномірно між всіма DNS-серверами мережі. Наприклад, у першому випадку DNS-сервер домена mmt.ru зберігатиме відображення для всіх імен, що закінчуються на mmt.ru: www1.zil.mmt.ru, ftp.zil.mmt.ru, mail.mmt.ru і т.д. У другому випадку цей сервер зберігає відображення тільки імен типу mail.mmt.ru, www.mmt.ru, а всі інші відображення мають зберігатися на DNS-сервері піддомена zil.

Кожний DNS-сервер крім таблиці відображень імен містить посилання на DNS-сервери своїх піддоменів. Ці посилання зв'язують окремі DNS-сервери в єдину службу DNS. Посилання є IP-адресами відповідних серверів. Для обслуговування кореневого домена виділено декілька дублюючих один одного DNS-серверів, IP-адреси яких є широко відомими (про них можна довідатися, наприклад, в InterNIC).

Процедура дозволу DNS-імені багато в чому аналогічна процедурі пошуку файловою системою адреси файла за його символьним іменем. Дійсно, в обох випадках складене ім'я відбиває ієрархічну структуру організації відповідних довідників – каталогів файлів або таблиць DNS. Тут домен і доменний DNS-сервер є аналогом каталогу файлової системи. Для доменних імен, так само як і для символьних імен файлів, характерна незалежність іменування від фізичного місця розташування.

Процедура пошуку адреси файла за його символьним іменем полягає в послідовному перегляді каталогів, починаючи з кореневого. При цьому попередньо перевіряється кеш і поточний каталог. Для визначення IP-адреси за доменним іменем також необхідно переглянути всі DNS-сервери, що обслуговують ланцюжок піддоменів, що входять в ім'я хоста, починаючи з кореневого домена. Істотною же відмінністю є те, що файлова система розташована на одному комп'ютері, а служба DNS по своїй природі є розподіленою.

Існують дві основні схеми дозволу DNS-імен. У першому варіанті роботу з пошуку IP-адреси координує DNS-клієнт:

- DNS-клієнт звертається до кореневого DNS-сервера із вказівкою повного доменного імені;
- DNS-сервер відповідає, вказуючи адресу наступного DNS-сервера, що обслуговує домен верхнього рівня, задану в старшій частині запитаного імені;
- DNS-клієнт робить запит наступного DNS-сервера, який відсилає його до DNS-сервера потрібного піддомена, і т.д., доки не буде знайдений DNS-сервер, у якому зберігається відповідність запитаного імені IP-адресі. Цей сервер дає остаточну відповідь клієнту.

Така схема взаємодії називається нерекурсивною або ітеративною, коли клієнт сам ітеративно виконує послідовність запитів до різних серверів імен. Оскільки ця схема завантажує клієнта досить складною роботою, то вона застосовується рідко.

У другому варіанті реалізується рекурсивна процедура:

- DNS-клієнт запитує локальний DNS-сервер, тобто той сервер, що обслуговує піддомен, до якого належить ім'я клієнта;
- якщо локальний DNS-сервер знає відповідь, то він відразу ж повертає його клієнту; це може відповідати випадку, коли запитане ім'я входить у той самий піддомен, що й ім'я клієнта, а також може відповідати випадку, коли сервер уже дізнавався про дану відповідність для іншого клієнта й зберіг його у своєму кеші;
- якщо ж локальний сервер не знає відповідь, то він виконує ітеративні запити до кореневого сервера й т.д. так само, як це робив клієнт у першому варіанті; отримавши відповідь, він передає його клієнту, який весь цей час просто чекав його від свого локального DNS-сервера.

У цій схемі клієнт передоручає роботу своєму серверу, тому схема називається непрямою або рекурсивною. Практично всі DNS-клієнти використовують рекурсивну процедуру.

Для прискорення пошуку IP-адрес DNS-сервери широко застосовують процедуру кешування відповідей, які проходять через них. Щоб служба DNS могла оперативно опрацьовувати зміни, які відбуваються в мережі, відповіді кешуються на певний час – звичай від декількох годин до декількох днів.

Висновки

- У стеку TCP/IP використовуються три типи адрес: локальні (що також називаються апаратними), IP-адреси й символічні доменні імена. Всі ці типи адрес привласнюються вузлам складеної мережі незалежно одна від одної.
- IP-адреса має довжину 4 байти й складається з номера мережі та номера вузла. Для визначення границі, що відокремлює номер мережі від номера вузла, реалізуються два підходи. Перший заснований на понятті класу адреси, другий – на використанні масок.
- Клас адреси визначається значеннями декількох перших біт адреси. В адресах класу А під номер мережі виділяється один байт, а інші три – під номер вузла, тому вони використовуються у самих великих мережах. Для невеликих мереж більш підходять адреси класу З, у яких номер мережі

займає три байти, а для нумерації вузлів може бути використаний тільки один байт. Проміжне положення займають адреси класу В.

- Інший спосіб визначення, яка частина адреси є номером мережі, а яка номером вузла, заснований на використанні маски. Маска – це число, що використовується в парі з IP-адресою; двійковий запис маски містить одиниці в тих розрядах, які в IP-адресі мають інтерпретуватися як номер мережі.

- Номери мереж призначаються або централізовано, якщо мережа є частиною Internet, або довільно, якщо мережа працює автономно.

- Процес розподілу IP-адрес по вузлах мережі може бути автоматизований за допомогою протоколу DHCP.

- Встановлення відповідності між IP-адресою й апаратною адресою (найчастіше MAC-адресою) здійснюється протоколом дозволу адрес ARP, який для цього переглядає ARP-таблиці. Якщо потрібна адреса відсутня, то виконується широкомовний ARP-запит.

- У стеку TCP/IP застосовується доменна система символічних імен, що має ієрархічну деревоподібну структуру, яка допускає використання в імені довільної кількості складових частин. Сукупність імен, у яких кілька старших складових частин збігаються, утворюють домен імен. Доменні імена призначаються централізовано, якщо мережа є частиною Internet, у протилежному випадку – локально.

- Відповідність між доменними іменами й IP-адресами може встановлюватися як засобами локального хосту з використанням файла hosts, так і за допомогою централізованої служби DNS, заснованої на розподіленій базі відображень «доменне ім'я – IP-адреса».

13.3 Контрольні запитання та завдання

1. Скільки рівнів має стек протоколів TCP/IP? Які їхні функції? Які особливості цього стека обумовлюють його лідируюче положення у світі мережних технологій?

2. Які протоколи стека TCP/IP належать до рівня Internet (рівня міжмережної взаємодії)?

3. Який протокол автоматизує процес розподілу IP-адрес по вузлах мережі?

4. Якими засобами встановлюється відповідність між доменними іменами й IP-адресами?

5. Які методи застосовуються для визначення границі, що відокремлює номер мережі від номера вузла?

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 4-е изд. / В.Г. Олифер, Н.А.Олифер. – СПб.: Питер, 2010. – 944 с.
2. Сети передачи данных. /Д. Бертсекас, Р. Галагер. – М.: Мир, 1989. – 544 с.
3. Комп'ютерні мережі: навчальний посібник / [О.Д. Азаров, С.М. Захарченко, О.В. Кадук та ін.] – Вінниця: ВНТУ, 2013. – 371 с.
4. Буров Є.В. Комп'ютерні мережі: підручник /Є.В. Буров. – Львів: «Магнолія 2006», 2010. – 262 с.
5. Кулаков Ю.О., Жуков І.А. Комп'ютерні мережі. Навчальний посібник / за ред. Ю.О. Кулакова. – К: НАУ, 2009. – 392 с.
6. Комп'ютерні мережі: [навчальний посібник] / А.Г. Микитишин, М.М. Митник, П.Д. Стухляк, В.В. Пасічник. – Львів: «Магнолія 2006», 2013. – 256 с.

Електронне навчальне видання

КОНСПЕКТ ЛЕКЦІЙ
з дисципліни
«ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ СИСТЕМИ»
частина І

для студентів усіх форм навчання спеціальності 125 «Кібербезпека»
за освітньою програмою «Безпека інформаційних комунікаційних систем»

Упорядник ХАЛІМОВ Геннадій Зайдулович

Відповідальний випусковий Г.З. Халімов
Редактор Б.П. Косіковська
Комп'ютерна верстка Л.Ю. Светайло

План 2019 (друге півріччя), поз. 31

Підп. до використання 21.06.2019

Формат pdf.

Обсяг даних 7,36 Mb

ХНУРЕ. Україна. 61166, Харків, просп. Науки, 14, E-mail: info@nure.ua

Підготовлено в редакційно-видавничому відділі ХНУРЕ
Свідоцтво суб'єкта видавничої справи ДК №1409 від 26.06.2003