МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Київський національний університет будівництва і архітектури

В.М. Вишняков

Принципи побудови комп'ютерних мереж

Рекомендовано вченою радою Київського національного університету будівництва і архітектури як навчальний посібник для студентів галузі знань 12 «Інформаційні технології»

Київ 2022

УДК 004.7
В55

Рецензенти*: Д.В. Ланде*, д-р техн. наук, професор,
Київський політехнічний інститут імені Ігоря Сікорського;
Р.С. Одарченко, д-р техн. наук, професор,
Національний авіаційний університет

.

Затверджено на засіданні вченої ради Київського національного університету будівництва і архітектури, протокол № 4 від 24 січня 2022 року.

Вишняков В.М.

В55 Принципи побудови комп'ютерних мереж**:** навч. посіб. / В.М. Вишняков. – Київ.: КНУБА, 2022. – 124 с.

ISBN 978-966-627-194-8

Розглянуто основні поняття та сучасні технології побудови комп’ютерних мереж. Особливу увагу надано розгляду можливостей сумісного використання в мережі Інтернет четвертої та шостої версій протоколу *IP*. Наведено дані про перспективні розробки та напрями розвитку програмно-технічних засобів комп’ютерних мереж.

Призначений для студентів факультету автоматизації та інформаційних технологій.

УДК 004.7

ISBN 978-966-627-194-8

ЗМІСТ

ПЕРЕДМОВА

Автор посібника "Принципи побудови комп'ютерних мереж" спирався на досвід роботи відділу мережевих інформаційних технологій Державного науково-дослідного інституту автоматизованих систем в будівництві (ДНДІАСБ), де працював весь цей час за сумісництвом.

Щиру подяку автор висловлює колегам по роботі у ДНДІАСБ за цінні зауваження під час підготовки рукопису, а саме завідувачу лабораторії технічного та програмного забезпечення інформаційних мереж Дмитру Тарасюку, провідному науковому співробітнику кандидату фізико-математичних наук Анатолію Вовку та провідному інженеру Євгену Бєлкіну.

ВСТУП

Цей посібник є підсумком роботи автора на посаді завідувача відділу мережевих інформаційних технологій у Державному науково-дослідному інституту автоматизованих систем в будівництві (ДНДІАСБ), починаючи з 1970-х років, та викладання курсу “Комп’ютерні мережі” на факультеті автоматизації та інформаційних технологій у Київському національному університеті будівництва і архітектури протягом 2001-2022 років.

Метою посібника є надання допомоги студентам та особам, які мають бажання ознайомитись з принципами побудови комп’ютерних мереж, набути базові знання у цій галузі.

Матеріал цього посібника розраховано на читача з мінімальним рівнем підготовки. Тут у доступній формі викладено сукупність знань, які необхідно засвоїти згідно з програмою даного курсу. Автором зроблено спробу позбавити посібник від застарілого та зайвого матеріалу і надати відповіді на всі найважливіші запитання, які виникали у студентів під час вивчення курсу.

Сьогодні найвагомішим результатом розвитку мережевих технологій є всесвітня мережа Інтернет, яка значно поліпшила такі важливі процеси людської діяльності як спілкування та розповсюдження знань. Зараз можна за лічені хвилини ознайомитись з розробками в будь-якій галузі знань і також швидко розповсюджувати інформацію про власні досягнення. За кожен рік кількість інформації, яка пересилається у мережі Інтернет, збільшується майже у двічі.

Чому з величезної кількості запропонованих технічних рішень для створення комп’ютерних мереж було обрано саме ті технології, на яких побудовано сучасні мережі? Які властивості цих технологій зіграли вирішальну роль у їх виборі? Як і в яких умовах відбувався і продовжує відбуватись цей вибір? Що являють собою сучасні мережеві технології? На всі ці запитання можна знайти відповіді у даному посібнику. Набуття цих знань має полегшити майбутнім фахівцям перехід від навчання до професійної діяльності, допомогти впроваджувати, створювати та використовувати мережеві технології.

Р О З Д І Л 1

КОНЦЕПЦІЇ КОМП'ЮТЕРНИХ МЕРЕЖ

1.1. Топологія комп’ютерних мереж та фізичних зв'язків

Загальне поняття топології мереж означає схему розміщення і з'єднання вузлів без врахування їх фізичних розмірів. Для топології комп'ютерних мереж крім того не має значення територіальне розміщення комп'ютерів, а враховується лише конфігурація їх логічних зв'язків. Топологію комп'ютерної мережі слід розуміти, як схему з'єднання вузлів без врахування відстані між ними і їх територіального розміщення.

Крім поняття топології комп'ютерної мережі (КМ) існує поняття, топології фізичних зв'язків КМ, куди включають у якості вузлів не тільки комп'ютери, але й з'єднувальне обладнання, наприклад, комутатори або концентратори. При цьому територіальне розміщення обладнання також не враховується. Головною ознакою, за якою легко відрізнити вузол КМ від з'єднувального обладнання, є наявність унікальної адреси, без котрої вузол не здатен забезпечити обмін даними. Найпростіші варіанти топології, які відповідають окремим (не поєднаним між собою) мережам, показані на рис.1.1 [1].

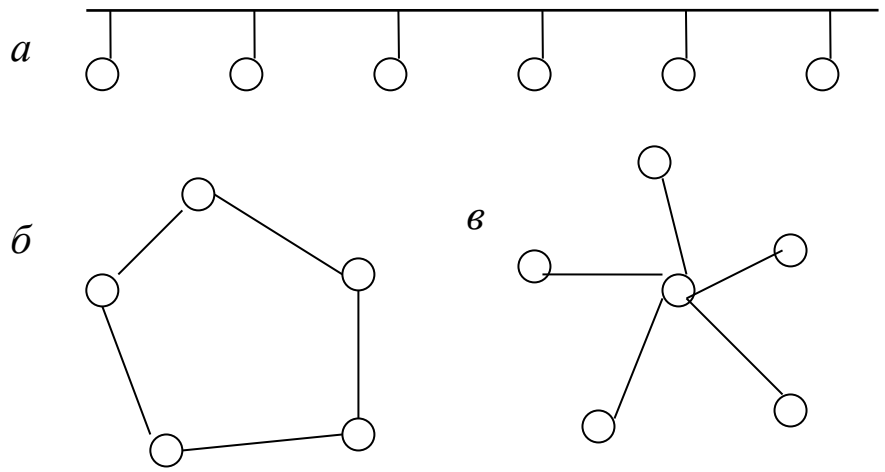


Рис. 1.1. Основні варіанти топології КМ:
а – шинна (спільна шина); *б* – кільцева ("кільце); *в* – зіркоподібна ("зірка").

Перші мережі з більшою за два кількістю комп'ютерів було створено за цими варіантами топології. У випадку мережі з двома вузлами усі ці варіанти будуть мати однаковий вигляд. Найбільш відомими і розповсюдженими у наш час є мережі, що побудовані за технологією *Ethernet*, у яких топологія є шинною.

Ця технологія заснована на ідеях, які було впроваджено у мережі *ALOHA* в Гавайському університеті ще у 1971 році. Це, по-перше, передавати дані частками у пакетах певної довжини із заголовком, де вказані адреси відправника та одержувача, як показано на рис. 1.2.

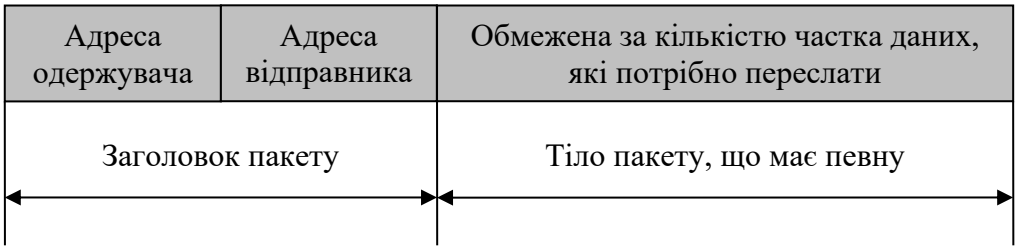


Рис. 1.2. Структура пакету даних.

По-друге, це те, що будь-який вузол може починати відправлення пакету у будь-який момент часу, використовуючи для цього єдину для усіх вузлів частоту радіо ефіру. Ідею щодо формування даних у вигляді пакетів було підтримано і впроваджено усіма розробниками майбутніх технологій побудови КМ. Але ідею щодо початку передавання даних будь-яким вузлом у будь-який момент визнав лише Роберт Меткалф, який прийняв цю ідею за основу технології *Ethernet.* Він казав, що ця технологія зможе у майбутньому стати єдиною технологією побудови КМ. Інших розробників налякала відправка пакетів у будь-який момент через можливість їх зіткнення у спільній шині, що приводить до знищення даних. Таку ситуацію прийнято називати **колізією**.

Найперше обладнання для створення КМ вийшло на ринок у 1983 році. У ньому було використано технологією *ARCNet* із топологією "зірка" Ця технологія була позбавлена колізій, бо центральний вузол надавав дозвіл на передачу іншим вузлам за допомогою відправлення спеціальних пакетів, які було названо **маркери**. Другу технологією з використанням маркерів, але з кільцевою топологією, було розроблено фірмою *IBM* під назвою *Token ring*. У 1984 році цю технологію було стандартизовано і вона у ті часи вважалась найкращою. Передача пакету даних за цією технологією проілюстрована на рис. 1.3.

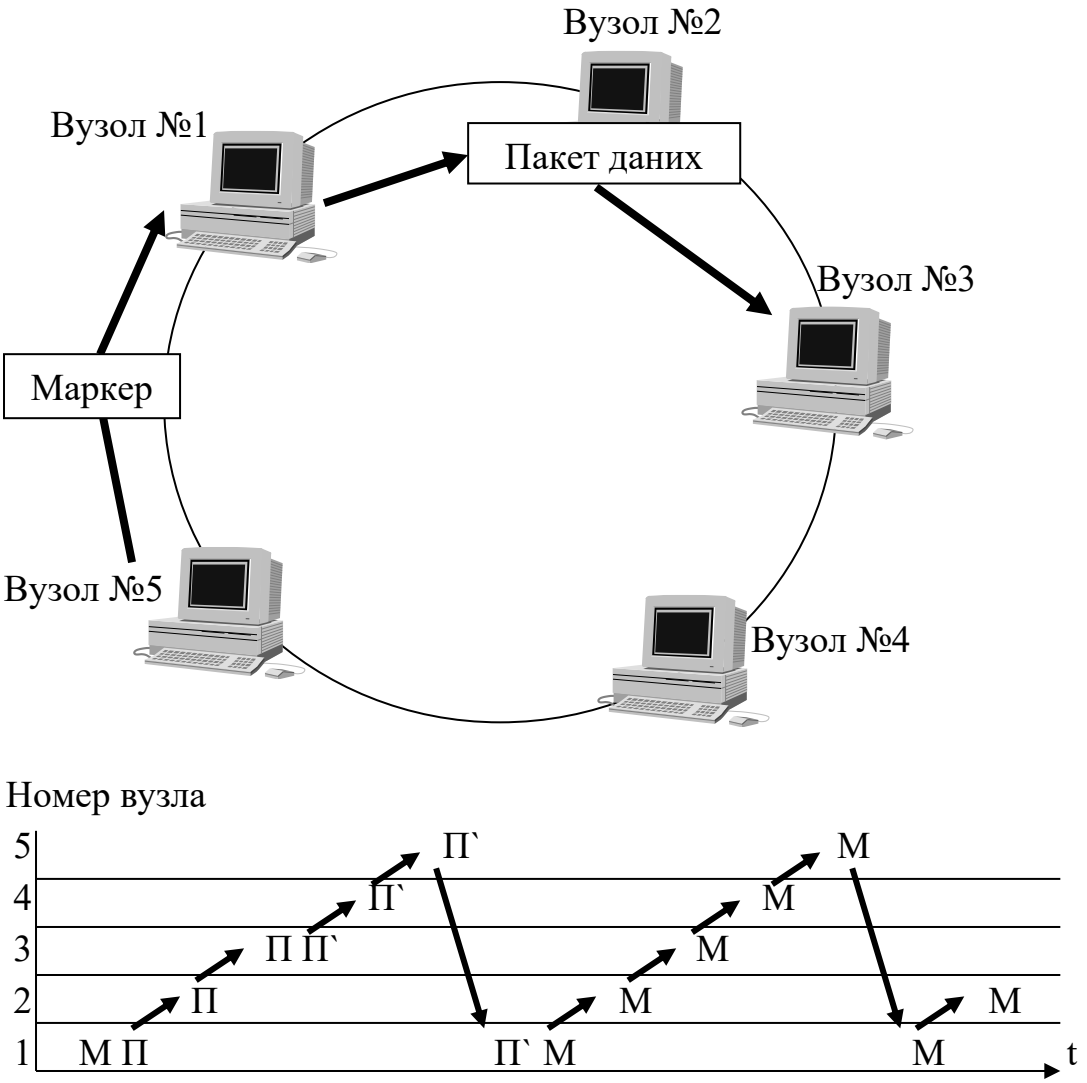


Рис. 1.3. Передача пакету даних за технологією *Token Ring*

Маркер передається по кільцю від вузла до вузла доки не потрапить на вузол, що має пакет даних для відправки. Отримавши маркер М, вузол №1 передає замість маркера пакет П. Вузол №2 не перехоплює пакет, а відсилає його далі по колу на вузол №3. Вузол №3, отримавши пакет на свою адресу, відправляє далі по колу копію пакета П з прапорцем про отримання П`. Пакет П` по колу повинен дійти до вузла №1, де його зміст порівнюється з пакетом даних, що було відправлено. У разі позитивного результату порівняння вузол №1 відправляє маркер М, який до цього часу був затриманий. Після цього рух маркера по колу відновлюється.

Ця технологія на перший погляд виглядає простою, але в реальних умовах виникають ускладнення через наступне.

- Комп’ютери у мережі можуть вимикатись і вмикатись.

- Завада може знищити маркер і треба буде його відновлювати.
- Маркер під час відновлення може подвоїтись.
- Слід передбачити можливість підключення нових комп’ютерів.

Цей самий алгоритм у 1987 році було застосовано у технології *FDDI*, яка за рахунок використання волоконно-оптичного кабелю на той час була найбільш швидкісною. У 1984 році розпочався продаж обладнання для мереж за технологією *Ethernet.* Порівняння характеристик технологій побудови КМ надано у табл. 1.1.

Таблиця 1.1.

Порівняння технологій побудови комп'ютерних мереж

Технологія	*ARCNet*	*Token Ring*	*FDDI*	*Ethernet*
Топологія КМ	Зірка	Кільце	Кільце	Шинна
Швидкість	20 Mb/s	16 Mb/s Gb/s з 2001р.	100 Mb/s	10/100/1000 Mb/s 40/100 Gb/s
Фізичне середовище	Кабель коаксіальний	Скручена пара	Оптичний кабель	Коаксіальний кабель, скручена пара, оптичний кабель
Кількість вузлів	255	260	1000	1024
Рік початку / закінчення підтримки	1983/1990	1984/2004	1987/1994	1984/ підтримка продовжується

З цієї таблиці бачимо що, минуло близько 20 років після закінчення підтримки тих технологій КМ, у яких було обрано зіркоподібну або кільцеву топологію. Вже більше ніж 15 років шинна топологія КМ є єдиною на ринку обладнання КМ. Це результат багаторічного досвіду користування різними варіантами технологій КМ. Виявилось, що з колізіями можна успішно боротись, а також завдяки збільшенню пропускної спроможності каналів у зв'язку з переходом на волоконну оптику ймовірність колізій значно зменшилась. Крім того, з появою комутаторів вдалося майже повністю позбутись колізій. Розглянемо основні етапи розвитку технології *Ethernet* для пояснення причин її перетворення від ненадійної до найкращої і єдиної на світовому ринку технологій КМ. Схема однієї с перших *Ethernet* мереж показана на рис. 1.4.

Рис. 1.4. Схема мережі *Ethernet* на коаксіальному кабелі

Працездатність *Ethernet* мереж на коаксіальному кабелі порушувалась досить часто. Пошкодження кабелю або засобів для з'єднання у будь якому місці призводило до відмови усієї мережі. Також під час приєднання або від'єднання абонентів слід було зупиняти роботу мережі. Суттєвим кроком щодо поліпшення роботи *Ethernet* мереж була поява концентраторів, які являли собою шину, що була захищена пластиковим корпусом. На рис. 1.5 надано вигляд одного з варіантів таких концентраторів.

Рис. 1.5. Концентратор (*HUB*) з комплектом засобів для з'єднання з коаксіальним кабелем (на передньому плані зліва направо розміщені: кінець кабелю з роз'єднувачем, Т- з'єднувач і термінатор)

Цей концентратор дозволяв поєднати шину на коаксіальному кабелі з внутрішньою шиною концентратора. Приєднання до останньої відбувалось за допомогою шнурів типу *Patch Cord* з роз'єднувачами типу *RJ*-45. Гнізда до цих шнурів почали встановлювати майже на усіх моделях комп'ютерів. Відмова від коаксіального кабелю значно покращала працездатність мереж *Ethernet*, бо пошкодження чи відключення кабелю на будь-якому комп'ютері не впливало на роботу інших комп'ютерів у мережі. Крім того, швидкість передавання даних

було підвищено до 100 Мбіт/с проти 10 Мбіт/с, як у разі використання коаксіального кабелю.

Наступним кроком щодо поліпшення роботи *Ethernet* мереж була поява комутаторів, які виконують ту ж функцію, що й концентратори, але мають пам'ять на кожному порту. У цій пам'яті можуть затримуватись пакети даних у разі зайнятості шини, що дозволяє уникати колізій. При цьому топологія фізичних зв'язків з появою з'єднувальних пристроїв таких, як концентратори та комутатори, буде зіркоподібною або деревоподібною (але не кільцевою), як показано на рис. 1.6.

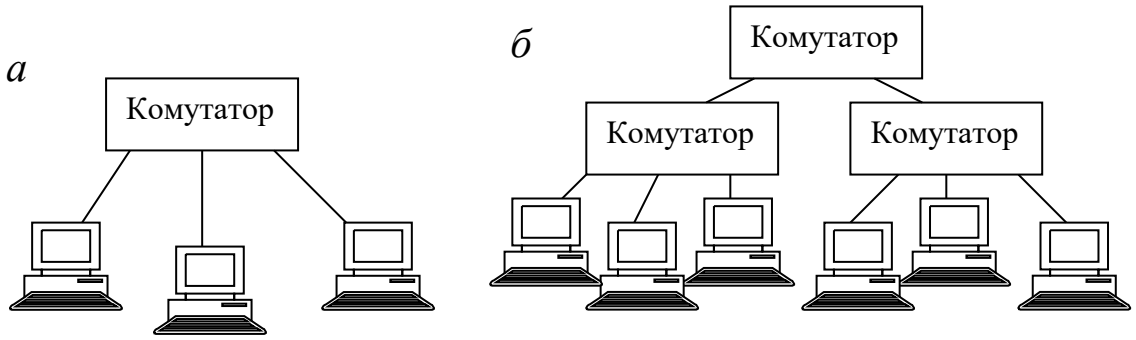


Рис. 1.6. Топологія фізичних зв'язків мережі *Ethernet*:
а – зіркоподібна; *б* – деревоподібна.

Впровадження комутаторів чи концентраторів не змінює топологію логічних зв'язків мережі, яка залишається шинною.

Крім топології окремих мереж покажемо топологію об'єднаних мереж, які найчастіше являють собою фрагмент мережі Інтернет. Об'єднання мереж відбувається за допомогою вузлів, які називають маршрутизаторами (*Router*). Кожен такий вузол має бути підключено до двох або більше окремих мереж. Він повинен мати адресу у кожній з цих мереж. Розробники мережі Інтернет виділяли у об'єднаних мережах усього два типи вузлів: **хост** (***Host***) і **маршрутизатор** (***Router***).

- ***Host*** (**хост**) – вузол, що відсилає та приймає пакети з даними.
- ***Router*** (**маршрутизатор**) – вузол, що пересилає пакети з однієї мережі в іншу.

Схему об'єднаної мережі, яка складається з декількох окремих мереж, зображено на рис.1.7, де жирною лінією позначені шини. Шиною можуть бути не тільки комутатори чи їх з'єднання, але й радіо ефір, як у мережі *ALOHA*, ідеї якої покладено у технологію *Ethernet*. Назва *Ethernet* походить від *Ether* (ефір) та network (мережа) на честь мережі *ALOHA*.

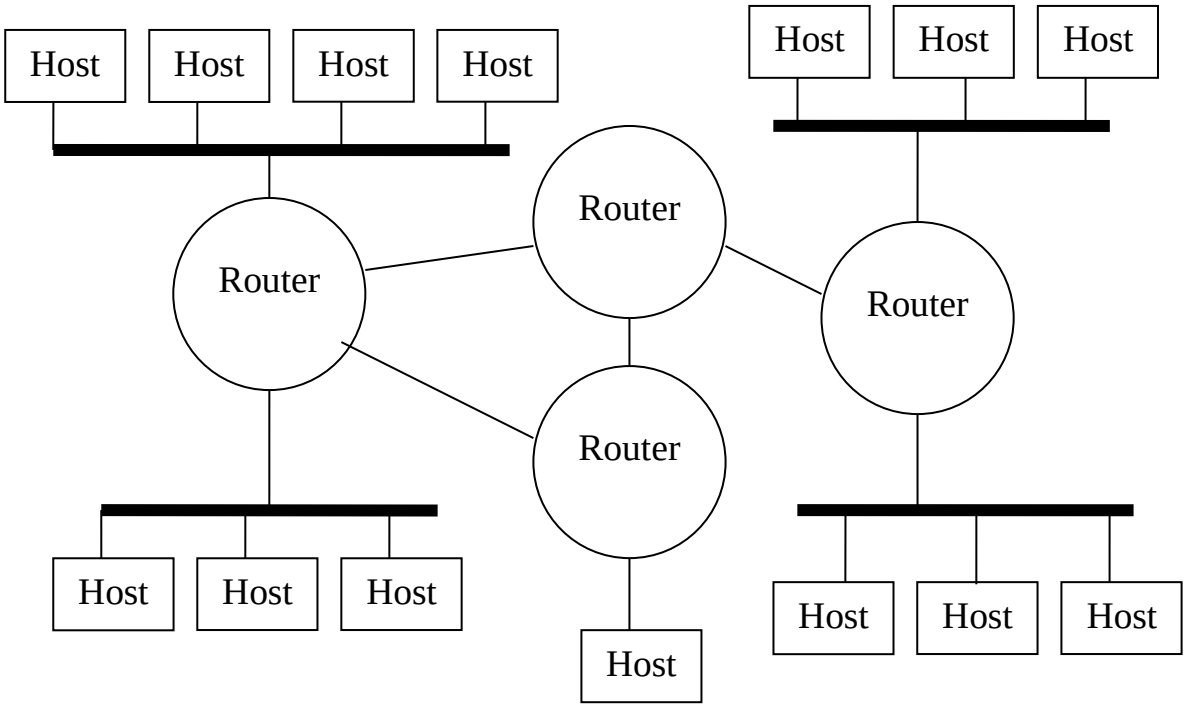


Рис. 1.7. Топологія складної мережі

Маршрутизатори можна з'єднувати між собою за будь-яким варіантом топології, включаючи кільця.

Таким чином, на топологію мережі не впливають фізичні розміри, як вузлів, так і з'єднань між ними. Також не впливає на топологію територіальне розміщення ані вузлів, ані з'єднувального обладнання. Єдине, що впливає на топологію, це те, як компоненти мережі з'єднані між собою. При цьому, на топологію КМ впливають лише зв'язки між вузлами, а у топологію фізичних зв'язків включають ще й з'єднувальне обладнання.

1.2. Масштаб комп'ютерних мереж

Поняття масштабу КМ (на мові оригіналу *geographic scale*) означає розмір території, на якій розміщено мережу. На початку створення КМ їх за масштабом розподіляли на локальні (***Local Area Network, LAN***) та глобальні (*Wide Area Network, WAN*). Локальні мережі є територіально обмеженими. Вони забезпечують зв'язок в межах від кімнати до групи сусідніх будинків. Глобальні мережі територіально не обмежені. Їх поділяють на дві наступні категорії:

- **магістральні мережі** (*Backbone*), що забезпечують зв'язок між віддаленими потужними вузлами різних міст, країн, континентів;

- **мережі доступу** (*Access network*), які забезпечують зв’язок між віддаленою мережею або віддаленим комп’ютером з деякою головною мережею.

Проміжне місце між локальними та глобальними мережами займають **регіональні мережі** (*Metropolitan Area Network, MAN*), що обслуговують територію міста, та **кампусні мережі** (*Campus Area Network, CAN* від англ. *campus* – університетське містечко).

Протягом останніх років з'явилось чимало нових позначень масштабу мереж. Крім поняття *geographic scale* вживають поняття *physical extent* або *spatial scope*. Всі ці поняття означають фізичний розмір мережі. Кожну нову технологію побудови КМ відносять до конкретного типу за масштабом. Для мереж мінімального масштабу у 2015 році було прийнято стандарт *IEEE P*1906.1 під назвою *Nanoscale and Molecular Communication Framework* (нанорозмірна та молекулярна комунікація). Таким мережам надали назву ***Nanoscale network*** (**нанорозмірна мережа**).

Для мереж, що утворюються для однієї людини та знаходяться поруч із нею надано назву ***Personal area network, PAN***. (**Персональна мережа**). Прикладом такої технології є *Bluetooth.*

Для мереж, що обслуговують людське тіло для проведення медичних процедур надано назву ***Body area network, BAN*** (**Мережа людського тіла**). Це можуть бути пристрої, що вбудовані в тіло як імплантати, можуть бути встановлені на поверхні тіла або можуть переноситись, наприклад, у кишенях одягу чи в спеціальних сумках.

Для мереж, що утворюються у житловому приміщенні (вдома) для зв'язку між комп'ютерами, мобільними та різними домашніми пристроями надано назву ***Home area network, HAN*** (**Домашня мережа**).

Для мереж, що займаються консолідованим збереженням даних надано назву ***Storage area network, SAN*** (**Мережа збереження даних**).

Для підприємств, які створюють приватні мережі з використанням каналів Інтернету, надано назву ***Enterprise private network*** (**Приватна мережа підприємства**).

Для підприємств, які створюють приватні мережі з використанням свого власного обладнання, включаючи лінії зв'язку, технологіям побудови таких мереж надано назву ***Virtual private network, VPN*** (**Віртуальна приватна мережа**).

Для мереж радіо доступу у різних варіантах стільникового зв'язку, які дозволяють отримувати послуги Інтернету, надано назву ***Radio access network, RAN*** (**Мережа радіо доступу**).

Для мереж, що використовуються для підтримки мобільних пристроїв у довільній кількості, зон супутникового покриття, а також бездротових мереж надано назву ***Global area network***, ***GAN*** (**Глобальна мережа**).

У табл. 1.2 надано перелік назв мереж за зростанням масштабу.

Таблиця 1.2.

Назви мереж за зростанням масштабу

Назва мережі на мові оригіналу	Характеристика мережі
Nanoscale network	Нанорозмірна та молекулярна комунікація
Body area network, BAN	Обслуговування людського тіла
Personal area network, PAN	Обслуговування однієї людини
Home area network, HAN	В межах житла (будинку чи квартири)
Local Area Network, LAN	Обмежена територія та кількість вузлів
Enterprise private network	В межах підприємства зі своїми каналами
Virtual private network, VPN	В межах підприємства на каналах Інтернету
Storage area network, SAN	Консолідоване збереження даних
Campus Area Network, CAN	В межах студентського містечка
Metropolitan Area Network, MAN	В межах міста або регіону
Wide Area Network, WAN	Між містами без обмежень на відстань
Radio access network, RAN	З радіо доступом до Інтернету
Global area network, GAN	Без обмежень на кількість мобільних вузлів

Інформація, що надана у цій таблиці може змінюватись, бо розвиток КМ неможливо передбачити на декілька років через постійні інноваційні зміни.

1.3. Телекомунікаційні протоколи

Правила пересилання даних між процесами однакового рівня на різних вузлах мережі прийнято називати телекомунікаційними протоколами або для скорочення просто **протоколами**. Множину протоколів, якої достатньо для створення мережі, називають **стек протоколів**. Протягом останніх 30 років найбільш визнаним є **стек *TCP/IP***, на якому побудована мережа Інтернет. Розробники цього стеку розподіляли протоколи на чотири рівні. До верхнього або **прикладного** рівня віднесено протоколи, які визначають форму відправки і отримання даних прикладними процесами. Під цими процесами розуміють дії

людей, які звертаються до мережі для отримання інформації або за інших своїх потреб, а також дії сервера, який обслуговує користувачів мережі відповідно до їхніх запитів. Усі протоколи цього рівня призначені для пересилання повідомлень або файлів різного типу та розміру. Кожне пересилання на прикладному рівні з боку відправника доповнюється заголовком, у якому вказується тип протоколу, адреса одержувача та інформація про те, що слід робити одержувачу після отримання даного повідомлення чи файлу. На наступному **транспортному** рівні, відбувається розподіл повідомлень або файлів, включаючи заголовок, на рівні частини (крім останньої, куди потрапляє залишок), які називають "**сегменти**". Це є початком формування пакетів. Довжина сегментів обирається таким чином, щоб кожен з них після доповнення наступними заголовками вміщувався у пакет *Ethernet*, де кількість байт не повинна перевищувати 1500. Кожен сегмент доповнюється заголовком, де вказують числові ідентифікатори відправника та одержувача, які називають "**порти транспортного рівня**". Ці порти доповнюють адресу вузла, що дозволяє на комп'ютері з єдиною адресою одночасно виконувати до 65534 задачі (порти 0 та 65535 означають, що відповідь не очікується та порт не визначено, відповідно). Далі сегменти відправляються на **міжмережевий** рівень, де вони доповнюються *IP*-заголовком з *IP*-адресами хостів відправника і одержувача. Таким чином, з кожного сегмента створюється пакет міжмережевого рівня, який називають дейтаграмою. Далі дейтаграми через мережевий інтерфейс відправляються одержувачу на вказаний порт транспортного рівня до конкретного прикладного процесу. На рівні мережевого інтерфейсу, який ще називають "**Канальний**", до кожного пакета додається заголовок канального рівня з фізичними адресами відправника і одержувача окремої мережі. Під окремою слід розуміти будь-яку мережу, що побудована за одною технологією. Шлях пакету може пролягати через низку окремих мереж, між якими є маршрутизатори. Кожен маршрутизатор під час пересилання пакету з однієї мережі до іншої відкидає заголовок канального рівня мережі, яку пакет залишає, та встановлює заголовок тої мережі, до якої цей пакет потрапляє. Послідовність переформування даних на різних рівнях стеку *TCP/IP* під час відправлення повідомлення у мережу Інтернет показано на рис. 1.8.

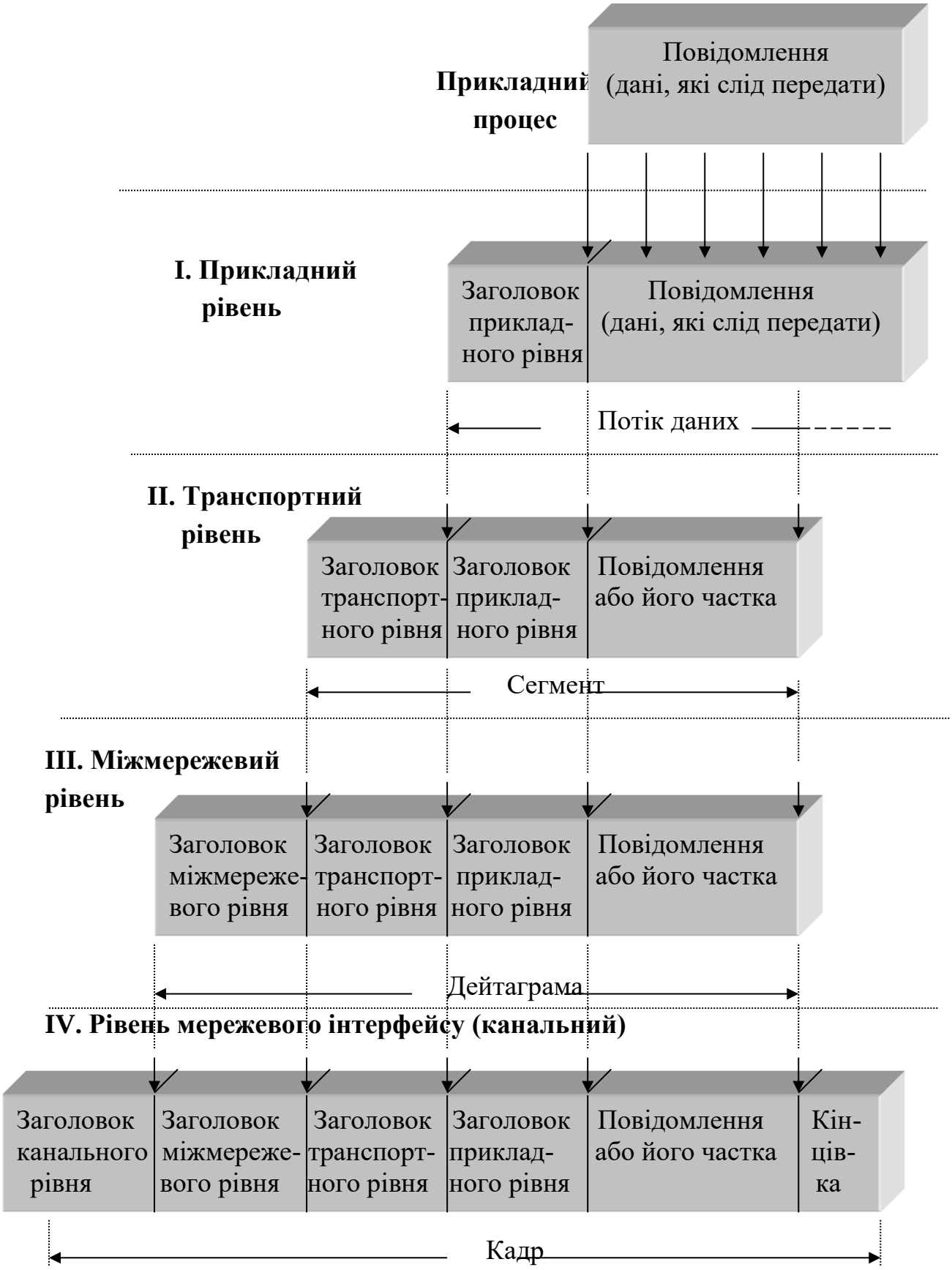


Рис. 1.8. Послідовність формування першого пакету на різних рівнях стеку протоколів *TCP/IP*

Назви та призначення основних протоколів стеку *TCP/IP* наведено у табл. 1.3, 1.4 та 1.5.

Таблиця 1.3.

Протоколи прикладного рівня стеку *TCP/IP*

Назва на мові оригіналу	Призначення
***FTP**, File Transfer Protocol*	Передача файлів (протокол застарілий)
***SSH**, Secure Shell*	Безпечна оболонка для обміну даними
***SFTP**, SSH File Transfer Protocol*	Безпечна передача файлів
***TFTP**, Trivial File Transfer Protocol*	Тривіальний протокол передачі файлів у безпечних локальних мережах
***HTTP,** Hypertext Transfer Protocol*	Передача гіпертекстових файлів без забезпечення захисту
***HTTPS,** Hypertext Transfer Protocol*	Захищена передача гіпертекстових файлів
***SMTP,** Simple Mail Transfer Protocol*	Передача електронної пошти до поштової скриньки
***POP3,** Post Office Protocol Version 3*	Отримання електронної пошти з поштової скриньки
***IMAP**, Internet Message Access Protocol*	Маніпулювання електронною поштою у поштовій скриньці, включаючи перегляд
***MGCP**, Media Gateway Control Protocol*	Передача голосових повідомлень, включаючи *IP*-телефонію
***MQTT**, Message Queuing Telemetry Transport*	Передача телеметрії, що набуває потреби з впровадженням Інтернету речей (*IoT*)
***NTP**, Network Time Protocol*	Синхронізація годинників зі звичайною точністю
***PTP,** Precision Time Protocol*	Синхронізація годинників з високою точністю
***RTSP**, Real Time Streaming Protocol*	Передача мультимедіа у реальному часі
***LDAP**, Lightweight Directory Access Protocol*	Пошук файлів у каталогах на сервері
***NNTP,** Network News Transfer Protocol*	Обмін повідомленнями між учасниками конференції та отримання новин
***SNMP**, Simple Network Management Protocol*	Керування мережевими пристроями (маршрутизаторами, комутаторами і т.ін.)
***SIP,** Session Initiation Protocol*	Встановлення сесій для обміну різними типами повідомлень
***TELNET**, **tel**etype **net**work*	Термінал управління сервером у мережі
***DHCP**, Dynamic Host Configuration Protocol*	Автоматичне надання параметрів для підключення до мережі (з *IP*-адресою)
***DNS**, Domain Name System*	Перетворення символьних адрес на *IP*-адреси серверів

Таблиця 1.4.

Протоколи транспортного рівня стеку *TCP/IP*

Назва на мові оригіналу	Призначення
***TCP**, Transmission Control Protocol*	Передача повідомлень або файлів з виявленням та виправленням помилок, що може затримувати доставку даних
***UDP**, User Datagram Protocol*	Передача повідомлень без виправлення помилок
***DCCP,** Datagram Congestion Control Protocol*	Передача повідомлень без виправлення помилок з відкиданням пакетів, що були затримані
***SCTP**, Stream Control Transmission Protocol*	Передача даних з захистом від *DDoS* атак та гарантуванням відсутності помилок, але зі збільшенням потрібних ресурсів
***RSVP**, Resource ReSerVation Protocol*	Резервування потрібної пропускної спроможності усіх маршрутизаторів на шляху пересилання даних
***QUIC,** Quick UDP Internet Connections*	Захищена шифруванням передача даних з виправленням помилок, що базується на декількох потоках за протоколом *UDP* та забезпечує більшу швидкість ніж *TCP*

Таблиця 1.5.

Протоколи міжмережевого рівня стеку *TCP/IP*

Назва на мові оригіналу	Призначення
***IP**, Internet Protocol*	Передача *IP*-пакетів від вузла відправника до вузла одержувача
***ICMP**, Internet Control Message Protocol*	Передача діагностичних повідомлень з використанням протоколу *IP*
***IGMP**, Internet Group Management Protocol*	Передача повідомлень для управління груповою адресацією з використанням протоколу *IPv*4

Розробку телекомунікаційних протоколів було розпочато у 1969 році в межах проекту *ARPANET* під керівництвом агенції *ARPA* Міністерства оборони США. Зараз назва цієї агенції *DARPA* (*Defense Advanced Research Projects Agency* – Агентство передових оборонних дослідницьких проектів). Перший протокол з

позначенням *RFC* 1 під назвою «Host Software» був написаний Стівом Крокером з Каліфорнійського університету в Лос-Анджелесі. З того часу усі розробки щодо розвитку мережі *ARPANET*, яка з часом перетворилася у Всесвітню мережу Інтернет, позначаються, як *RFC* (*Request for Comments* – пропозиція для обговорення) із черговим номером. На початок 2000 року було опубліковано *RFC* 2598, а на початок 2020 року – *RFC* 8700. Сучасний розвиток мережі Інтернет базується на пропозиціях, які можуть надавати усі бажаючі за встановленою формою. Для розгляду цих пропозицій у 1986 році створили відкрите міжнародне співтовариство *IETF* (*Internet Engineering Task Force* – Сили інженерної підтримки Інтернету). У цьому товаристві робота здійснюється у чисельних робочих групах, що займаються конкретною тематикою і налічують десятки тисяч фахівців. До 1993 року діяльність *IETF* підтримував уряд США, а у 1993 році була створена міжнародна професійна організація *ISOC* (***I**nternet **Soc**iety* – Інтернет товариство), яка зараз підтримує діяльність *IETF* і формує політику Інтернету. Своєю місією *ISOC* вважає «забезпечення відкритого розвитку, еволюції та використання Інтернету на благо людей у всьому світі». *ISOC* займається ще й питаннями освіти в країнах, що розвиваються, професійної підготовки фахівців, управління і координації різних проектів, які пов'язані з мережею Інтернет [2].

До появи на ринку стеку *TCP/IP*, який зараз визнано за стандарт у мережі Інтернет, було створено і стандартизовано у 1980 році стек ISO/OSI, у якому протоколи розподілялись на 7 рівнів, як показано у табл. 1.6.

Таблиця 1.6.

Ієрархічні рівні та протоколи стеку *ISO/OSI*

Номер та назва рівня	Протоколи
7. *Application* (Прикладний)	*FTAM, X*.400, *X*.500, *DAP*, *ROSE*, *RTSE*, *ACSE*, *CMIP*
6. *Presentation* (Презентаційний)	*ISO/IEC* 8823, *X*.226, *ISO/IEC* 9576-1, *X*.236
5. *Session* (Сесійний)	*ISO/IEC* 8327, *X*.225, *ISO/IEC* 9548-1, *X*.235
4. *Transport* (Транспортний)	*ISO/IEC* 8073, *TP*0, *TP*1, *TP*2, *TP*3, *TP*4 (*X*.224), *ISO/IEC* 8602, *X*.234
3. *Network* (Мережевий)	*ISO/IEC* 8208, *X*.25 (*PLP*), *ISO/IEC* 8878, *X*.223, *ISO/IEC* 8473-1, *CLNP X*.233, *ISO/IEC* 10589, *IS-IS*
2. *Data link* (Канальний)	*ISO/IEC* 7666, *X*.25 (*LAPB*), *X*.25 (*LAPB*), *Token Bus*, *X*.222, *ISO/IEC* 8802-2, *LLC* (*type* 1/2)
1. *Physical* (Фізичний)	*X*.25 (*X*.21*bis*), *EIA/TIA*-232, *EIA/TIA*-449, *EIA*-530, *G*.703

З 1983 року розпочалися суперечки між тими, хто захищав стандартний стек *ISO/OSI*, і тими, хто не зважаючи на стандарти, впроваджував у своїх мережах стек *TCP/IP*. У 1990 році на ринку з'явилась технологія *ATM*, яку деякі фахівці прогнозували як можливу заміну стеків *TCP/IP* та *ISO/OSI*. У 1991 році технологія *ATM* швидко набувала розвитку, але прогнози щодо неї виявились хибними. У 2009 році підтримку залишків технології *ATM* було остаточно припинено. Стек *ISO/OSI* з 1990-х років залишався лише у теорії, бо його практичне застосування не спостерігалось. З чисельних протоколів, що були розроблені за цим стандартом (див. табл. 1.6), крім протоколу *IS-IS*, жоден більше не підтримувався. Протокол *IS-IS* на деяких маршрутизаторах і зараз ще можна зустріти, але замість нього більшість використовує новіший протокол маршрутизації *OSPF* (*Open Shortest Path First* - Першим відкриває найкоротший шлях).

Для того, щоб з'ясувати чому стандарт *ISO/OSI* отримав поразку, слід пригадати, що у наступні декілька років після розробки стеку *ISO/OSI*, комп'ютерна техніка суттєво змінились. Спочатку вважали, що обмін даними між комп'ютерами буде незначним і для нього вистачить швидкості телетайпа, а для усіх майбутніх мереж достатньо стандартного стеку *ISO/OSI*. Презентаційний рівень був потрібен через те, що значна частина діючих комп'ютерів мала архітектуру *IBM System*/360 із застарілим кодом *EBCDIC* (*Extended Binary Coded Decimal Interchange Code* - розширений двійково-десятковий код обміну). Код *EBCDIC* мав шість несумісних між собою версій. Зараз комп'ютери такого типу називають динозаврами *IBM*, бо ці велетні займали цілий поверх, витрачаючи чимало енергетичних та інших ресурсів. У період з 1992 до 2000 років їх перетворили на брухт. У нових моделях комп'ютерів використовували стандартний код *ASCII* (*American standard code for information interchange*). Оскільки для обміну текстами між новими і старими комп'ютерами необхідно було робити перекодування, чим пояснюється потреба у презентаційному рівні. Після утилізації динозаврів *IBM* потреба у спеціальному презентаційному рівні зникла. Сесійний рівень являв собою гальма, які заважали підвищити швидкість передавання термінових команд управління. Слід нагадати, що згідно стандарту *ISO/OSI* кожен з семи рівнів був обов'язковим, що не дозволяло відкидати непотрібні рівні. Таким чином, користь, що залишилась у спадок від стеку *ISO/OSI*, це загальний принцип розподілу протоколів на ієрархічні рівні і деталізація рівня мережевого інтерфейсу шляхом його розподілу на фізичну і логічну частину. У сучасних стеках протоколів те, що стосується фізики приєднання до середовища передавання сигналів,

відокремлюють у фізичний рівень, а на канальному рівні залишають тільки логічні перетворення.

Провідні розробники мереж вважають, що стандартизувати треба лише протоколи, а формування стеків не потребує стандартизації. Для переважної більшості користувачів Інтернету 2000-х років стек протоколів налічував 5 рівнів, що наведені у табл. 1.7.

Таблиця 1.7.

Найпопулярніший стек протоколів у мережі Інтернет 2000-х років

Протокол	Назва рівня
HTTP	*Application* (Прикладний)
TCP	*Transport* (Транспортний)
IP	*Network* (Мережевий)
Ethernet	*Data link* (Канальний)
IEEE 802.3*ab*	*Physical* (Фізичний)

Хоч у цій таблиці вказано лише 5 протоколів, але цього достатньо для надання доступу до гіпертекстових сторінок, що цілком задовольняло групу користувачів. При цьому мережа може бути працездатною, а це означає, що вказаний набір протоколів відповідає поняттю стеку протоколів. Не можна визначити наперед повну кількість послуг, яку буде надавати мережа Інтернет, бо весь час з'являються нові послуги разом з новими протоколами. Також неможливо визначити наперед потреби користувачів, бо ці потреби постійно зростають і їх складно передбачити. Наприклад, для персональних комп'ютерів типу *IBM PC* був розроблений стек з чотирьох рівнів, що показані у табл. 1.8.

Таблиця 1.8.

Стек протоколів *NetBIOS/SMB*

Назва протоколу	Призначення
SMB, Server Message Block	Забезпечення прикладним процесам доступу до файлів та принтерів в межах мережі
NetBIOS, Network Basic Input/Output System	Доповнення базової операційної системи *IBM PC* функціями роботи у мережі
Ethernet	Утворення каналу передачі даних
IEEE 802.3-2002	Утворення фізичного зв'язку між вузлами

Цей стек протоколів призначений для групової роботи з використанням спільного принтера та обміну файлами в межах однієї локальної мережі. Для того, щоб розширити можливості групової роботи з виходом за межі локальної

мережі можна скористатись протоколами мережі Інтернет. Для цього доповнюють стек *NetBIOS/SMB* двома протоколами зі стеку *TCP/IP* і отримують об'єднаний стек, який показано у табл. 1.9. Це дозволяє розширити групову роботу в межах мережі Інтернет, але для цього необхідно ще встановити на кожен вузол реальну *IP* адресу.

Таблиця 1.9.

Стек протоколів *NetBIOS/SMB* над стеком *TCP/IP*

Назва протоколу	Призначення
SMB, Server Message Block	Забезпечення прикладним процесам доступу до файлів та принтерів в межах мережі
NetBIOS, Network Basic Input/Output System	Доповнення базової операційної системи *IBM PC* функціями роботи у мережі
TCP	Виконання функцій транспортного рівня
IP	Виконання функцій міжмережевого рівня
Ethernet	Утворення каналу передачі даних
IEEE 802.3-2002	Утворення фізичного зв'язку між вузлами

Наведені приклади показують, що можна формувати стеки з існуючих протоколів для різних застосувань без прив'язки до конкретної кількості рівнів. Таким чином, можна погодитись з фахівцями, що вважають стандартизацію на рівні протоколів цілком достатньою, а вводити стандарти, які обмежують можливість маніпулювання вибором протоколів для стеків не є доцільним.

1.4. Принципи узгодження взаємодії протоколів

Функціонування програмного забезпечення на кожному з рівнів стеку може відбуватись за різними протоколами та у різних режимах. Це залежить від типу обладнання, його налаштування та від ініціатора з'єднання. Головне щоб у кожному з'єднанні між відправником і одержувачем даних на усіх рівнях стеку було досягнуто узгоджену взаємодію.

Розглянемо як це відбувається на фізичному та канальному рівнях, які необхідні для функціонування усіх стеків протоколів.

Найнижчий фізичний рівень охоплює характеристики приєднання до фізичного середовища передавання сигналів. На цьому рівні визначаються типи роз’єднувачів та антен, а також такі фізичні характеристики сигналів, як потужність і частота. На канальному рівні визначають правила формування

пакетів у вигляді послідовності біт, а саме формати заголовків та кінцівок кадрів. Нагадаємо, що кадрами називають пакети канального рівня.

Апаратура, що реалізує фізичний та канальний рівні має низку підрівнів, що показані на рис.1.9.

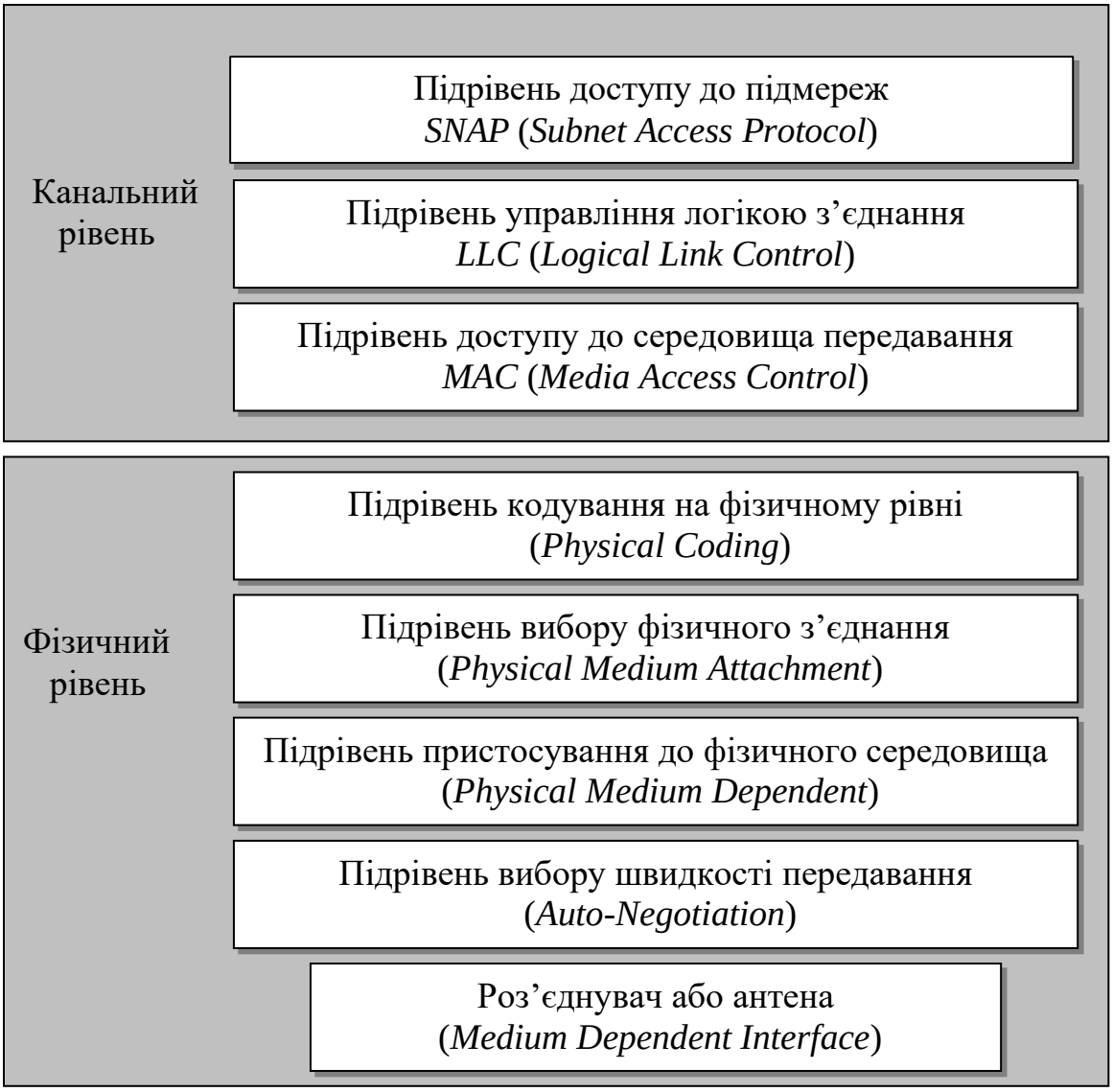


Рис. 1.9. Підрівні канального та фізичного рівнів

Наявність підрівнів пояснюється тим, що у більшості засобів доступу до мережі передбачена можливість використання різних варіантів технологій фізичного та канального рівнів. Ці технології стандартизуються підрозділом Інституту інженерів з електротехніки та електроніки *IEEE* (*Institute of Electrical and Electronics Engineers*). Стандарти фізичного та канального рівнів розробляє група *IEEE* 802. Вона готує стандарти для кабельних *Ethernet* мереж (*IEEE* 802.3), а також для бездротових *Wi-Fi* (*IEEE* 802.11), *WiMAX* (*IEEE* 802.16) та *WRAN* (*IEEE* 802.22) мереж. Перелік цих стандартів надано у табл. 1.10.

Таблиця 1.10.

Стандарти для фізичного та канального рівнів КМ

Код стандарту/рік	Опис
IEEE 802.3/1983	10 Мбіт/с через товстий коаксіальний кабель
IEEE 802.3*a*/1985	10 Мбіт/с через тонкий коаксіальний кабель
IEEE 802.3*i*/1990	10 Мбіт/с через скручену пару кат. 3
IEEE 802.3*j*/1993	10 Мбіт/с через оптичне волокно
IEEE 802.3*u*/1995	10/100 Мбіт/с (сумісність з *IEEE* 802.3і)
IEEE 802.3*z*/1998	1000 Мбіт/с через оптичне волокно
IEEE 802.3*ab*/1999	1000 Мбіт/с через скручену пару кат. 5E
IEEE 802.3-2002/2002	Включає в себе всі попередні стандарти
IEEE 802.3*ae*/2003	10 Гбіт/с через оптичне волокно
IEEE 802.3*an*/2006	10 Гбіт/с через скручену пару кат. 6 або 7
IEEE 802.11*a*/1999	до 54 Мбит/с на частоті 5 ГГц
IEEE 802.11*b*/1999	до 11 Мбит/с на частоті 2,4 ГГц
IEEE 802.11*g*/2003	до 54 Мбит/с на частоті 2,4 ГГц
IEEE 802.11*n*/2008	до 600 Мбит/с на частоті 2,4/5 ГГц
IEEE 802.11*ac*/2014	до 6933 Мбит/с на частоті 5 ГГц
IEEE 802.11*ax*/2020	до 9608 Мбит/с на частоті 2,4/5/6 ГГц
IEEE 802.16*n*/2013	до 100 Мбит/с мобільний до 1 Гбит/с фіксований
IEEE 802.16-2017/2017	Об'єднує останні стандарти на частотах 2-66 ГГц
IEEE 802.22.2/2011	до 18 Мбит/с для сільської місцевості

Кожен з цих стандартів об'єднує фізичний і канальний рівні, що були запропоновані у моделі *ISO/OSI*. У нових стандартах ці рівні складаються з низки підрівнів. Підрівні фізичного та канального рівнів являють собою спеціалізовані протоколи, які дозволяють пристроям домовлятись між собою щодо вибору найкращого режиму роботи в умовах появи чи зміни фізичного з’єднання. Дії за цими протоколами починаються одразу, як тільки з'являється фізичне середовище для передавання сигналів між вузлом та мережею. У разі підключення будь якого вузла одразу починається пошук для нього найбільш швидкісного та надійного з'єднання. Якщо є можливість приєднання через кабель або радіо-ефір, то найчастіше обирається кабель. Розвиток технологій канального рівня має тенденцію до об'єднання усіх раніше стандартизованих технологій так, щоб із новою технологією забезпечувалась можливість використання старих стандартів. Це зручно для користувачів, бо позбавляє від проблем у разі поновлення засобів доступу до мережі. Таким чином розв'язується

задача утворення каналу для передавання пакетів даних вищих рівнів усіх стеків протоколів. На кожному наступному рівні узгодження досягається завдяки наявності у заголовках пакетів інформації про тип протоколу, який потрібен для обробки даних, що пересилаються у пакеті. Наприклад, у кадрі *Ethernet*, формат якого представлено на рис. 1.10, для цього є поле *T* (*Type*).

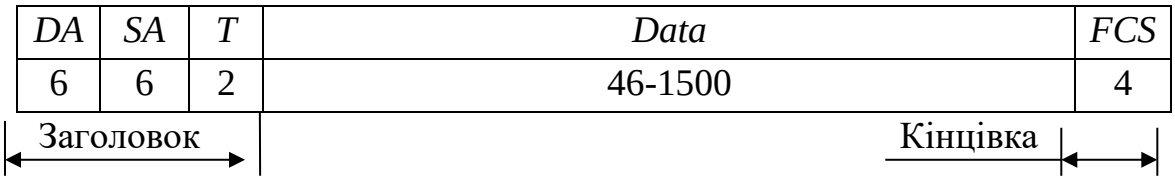


Рис. 1.10. Формат кадру *Ethernet*

Під позначкою кожного поля кадру наведено довжину у байтах.

DA – Destination Address – адреса одержувача.

SA – Source Address – адреса відправника.

T – Type – тип протоколу, пакет якого пересилають в полі даних.

Data – поле даних, які пересилаються у цьому кадрі.

FCS – Frame Check Sequence – контрольна сума.

У ранньому варіанті кадру *Ethernet*, який був призначений виключно для мереж компанії *Novell* (стек *IPX*/*SPX*), у полі даних пересилались лише пакети протоколу *IPX*. Тоді на місці поля *T* знаходилось поле *L* (*Length* – довжина поля даних у байтах). Оскільки значення *L* не може перевищити 1500, то значення *T* вирішили обрати більшими за 1500, щоб неможливо було переплутати кадри мережі компанії *Novell* з іншими мережами. Найчастіше у мережі Інтернет у полі *Data* пересилається пакет *IP* або *ARP*, для яких значення *T* дорівнює 2048 та 2054 відповідно. З переліком значень *T* можна ознайомитись за посиланням *https://en.wikipedia.org/wiki/EtherType.*

На початку заголовку кадру є адреси одержувача (*DA*) та відправника (*SA*). Ці адреси називають фізичними або апаратними, а найчастіше – *MAC*-адресами. Скорочення *MAC* означає *Media Access Control* (управління доступом до середовища). Незважаючи на велику кількість фірм, що виробляють обладнання мереж *Ethernet*, не може бути двох виробів з однаковими апаратними адресами. Про це піклується комітет 802 *IEEE*, який призначає кожному виробникові унікальний ідентифікатор організації *OUI* (*Organizationally Unique Identifier*). Адреса для кожного комп’ютера або адаптера чи іншого пристрою, у трьох перших (старших) байтах містить *OUI*, а у трьох правих (молодших) – номер виробу, що надає виробник. Перші два біти OUI завжди нульові.

Приклади різних *MAC*-адрес зображено на рис. 1.11.

Індивідуальна адреса, що містить *OUI* від комітету 802 *IEEE*

05	20	4D	40	30	A2

←*OUI*→

Індивідуальна адреса, що не містить *OUI* (для локальних дій)

40	00	07	00	30	A2

Широкомовна адреса пакета, що призначений усім вузлам мережі

FF	FF	FF	FF	FF	FF

Адреса пакета, що призначений групі вузлів,
які спеціально запрограмовані на прийняття пакетів з цією адресою

80	00	00	00	00	05

Рис. 1.11. Приклади різних варіантів *MAC*-адрес

Значення *MAC*-адрес у системах *Windows* прийнято відображати у вигляді 00-20-4D-40-30-A2, а у *UNIX* подібних операційних системах ця ж адреса буде відображена у формі 00:20:4d:40:30:a2.

Адреса відправника може бути тільки індивідуальною, а адреса одержувача може бути як індивідуальною, так і широкомовною або груповою.

Для групових (*multicast*) та широкомовних (*broadcast*) адрес старший біт першого байта завжди дорівнює 1. Групові адреси надають можливість призначати групу одержувачів пакета в межах своєї мережі, а широкомовні означають, що всі комп’ютери даної мережі повинні прийняти цей пакет.

Для індивідуальних адрес обмеженого використання (наприклад, для експериментів), старші два біти першого байта повинні мати значення 01.

Перетворення *IP* адреси одержувача у *MAC* адресу в кожній мережі, крізь яку пересилається *IP*-пакет, продовжується до знаходження вузла з потрібною *IP* адресою. Після того, як вузол з потрібною адресою знайдено, зміст пакету для обробки слід передати одному з протоколів вищого рівня даного вузла. Для цього у заголовку *IP* пакету (його структуру надано у табл. 1.12) вказано номер протоколу вищого рівня, до якого слід передати цей пакет для подальшої обробки.

Таблиця 1.12

Структура заголовку пакета *IPv4*

Найменування параметру	Кількість біт	Значення параметру
Номер версії	4	0100 (версія 4)
Довжина заголовка у 32-бітних словах	4	0101, що означає 20 байт
TOS Тип сервісу (*Type Of Service*)	3 1 1 1 1 1	Пріоритет від 000 до 111 (111 – вищій) 1 – мінімізувати затримку передавання 1 – максимізувати пропускну здатність 1 – максимізувати надійність 1 – мінімізувати вартість передавання 11111 – максимізувати безпечність
Загальна довжина пакета у байтах	16	Для мереж сім'ї *Ethernet* 1500 байт. Не може бути більше ніж 65500 байт
Ідентифікатор для фрагментації	16	Всі фрагменти одного пакета мають однакове значення цього ідентифікатора
Зарезервований біт Прапорець *DF* Прапорець *MF* Зміщення	1 1 1 13	Завжди нульовий 1 – заборона фрагментації пакета 1 – цей фрагмент не останній Кількість байт від початку поля даних
Час існування *TTL* (*Time To Live*)	8	Кількість вузлів, що може пройти пакет до моменту його знищення
Номер протоколу вищого рівня	8	1 – *ICMP*, 6 – *TCP*, 17 – *UDP*
Контрольна сума заголовка	16	Цю суму перераховують на кожному вузлі після зменшення *TTL*
IP-адреса відправника пакета	32	
IP-адреса одержувача пакета	32	

Стандартні номери надані усім протоколам, які можуть отримувати дані з *IP* пакетів. Повний перелік цих номерів можна отримати за посиланням https://en.wikipedia.org/wiki/List_of_IP_protocol_numbers. У цьому переліку надані значення номерів з 0 до 255. Значення початкової частини номерів протоколів наведено у табл. 1.13.

Таблиця 1.13

Значення номерів протоколів

Номер	**Назва протоколу**	***RFC***
0	*HOPOPT (IPv6 Hop-by-Hop Option)*	*RFC* 8200
1	*ICMP (Internet Control Message Protocol)*	*RFC* 792
2	*IGMP (Internet Group Management Protocol)*	*RFC* 1112
3	*GGP (Gateway-to-Gateway Protocol)*	*RFC* 823
4	*IP-in-IP (encapsulation)*	*RFC* 2003
5	*ST (Internet Stream Protocol)*	*RFC* 1190, *RFC* 1819
6	*TCP (Transmission Control Protocol)*	*RFC* 793
7	*CBT (Core-based trees)*	*RFC* 2189
8	*EGP (Exterior Gateway Protocol)*	*RFC* 888
9	*IGP (Interior Gateway Protocol)*	
10	*BBN RCC Monitoring*	
11	*NVP-II (Network Voice Protocol)*	*RFC* 741
12	*PUP (Xerox PUP)*	
13	*ARGUS*	
14	*EMCON*	
15	*XNET (Cross Net Debugger)*	
16	*CHAOS*	
17	*UDP (User Datagram Protocol)*	*RFC* 768
18	*MUX (Multiplexing)*	

Протоколи, номери яких вказуються у заголовку *IP* пакету, найчастіше є протоколами транспортного рівня *TCP* або *UDP*. Їх задача – переслати прийняті дані на прикладний рівень. Щоб дізнатись якому саме протоколу прикладного рівня відправляються дані використовується порт одержувача, номер якого завжди вказують у заголовку протоколу транспортного рівня так, як показано у табл. 1.14.

Таблиця 1.14

Структура *UDP*-заголовка

Найменування параметру	Кількість біт	Значення параметру
Порт відправника (*Source port*)	16	Номер обирається автоматично (будь який з не зайнятих
Порт одержувача (*Destined port*)	16	Номер означає протокол прикладного рівня (123 – *NTP*, 161 – *SNMP*)
Довжина пакету	16	Кількість байт даних та у *UDP*-заголовку
Контрольна сума	16	Може не формуватись

Номери портів, що закріплені за більшістю протоколів можна отримати з https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers, а деякі найбільш відомі з них наведено у табл. 1.15.

Таблиця 1.15

Номери портів, що закріплені за найбільш відомими протоколами

Номер порту, назва заголовку	Протокол прикладного рівня, за яким закріплено порт
20 та 21, *TCP*	***FTP**, File Transfer Protocol*
22, *TCP* та *UDP*	***SSH**, Secure Shell* (Безпечна оболонка)
22, *TCP* та *UDP*	***SFTP**, SSH File Transfer Protocol*
23, *TCP* та *UDP*	***TELNET**, **tel**etype **net**work*
25, *TCP* та *UDP*	***SMTP,** Simple Mail Transfer Protocol*
53, *TCP* та *UDP*	***DNS**, Domain Name System*
67, *TCP* та *UDP*	***DHCP**, Dynamic Host Configuration Protocol* (сервер)
68, *TCP* та *UDP*	***DHCP**, Dynamic Host Configuration Protocol* (клієнт)
69, *TCP* та *UDP*	***TFTP**, Trivial File Transfer Protocol*
80, *TCP* та *UDP*	***HTTP,** Hypertext Transfer Protocol*
110, *TCP* та *UDP*	***POP3,** Post Office Protocol Version 3*
119, *TCP* та *UDP*	***NNTP,** Network News Transfer Protocol*
123, *TCP* та *UDP*	***NTP**, Network Time Protocol*
143, *TCP* та *UDP*	***IMAP**, Internet Message Access Protocol*
161, *TCP* та *UDP*	***SNMP**, Simple Network Management Protocol*
319, *UDP*	***PTP,** Precision Time Protocol* (*Event Message*)
320, *UDP*	***PTP,** Precision Time Protocol* (*General Message*)
389, *TCP* та *UDP*	***LDAP**, Lightweight Directory Access Protocol*
443, *TCP* та *UDP*	***HTTPS,** SSH Hypertext Transfer Protocol*
554, *TCP* та *UDP*	***RTSP**, Real Time Streaming Protocol*
2427, *TCP* та *UDP*	***MGCP**, Media Gateway Control Protocol*
5060, *TCP* та *UDP*	***SIP,** Session Initiation Protocol*
8883, *TCP*	***MQTT**, Message Queuing Telemetry Transport*

Таким чином узгодження взаємодії протоколів між рівнями стеку *TCP*/*IP* відбувається завдяки тому, що у всіх заголовках пакетів є номер, який вказує на наступний протокол вищого рівня. На канальному рівні це поле *Type*, де у більшості випадків вказується число 2048, що означає *IP* протокол. У *IP* заголовку це є поле, де вказано номер протоколу вищого рівня, а у пакетах транспортного рівня це номер порту одержувача. На найвищий прикладний рівень вузла одержувача потрапляє повідомлення чи потік даних у такому ж вигляді, як на вході прикладного рівня вузла відправника.

В и с н о в к и

1. Поняття топології мережі означає схему розміщення і з'єднання вузлів без врахування їх розмірів. Для топології комп'ютерної мережі (КМ) не має значення територіальне розміщення комп'ютерів, а враховується лише конфігурація їх логічних зв'язків. Топологію комп'ютерної мережі слід розуміти, як схему з'єднання вузлів без врахування відстані між ними і їх територіального розміщення.
2. Крім поняття топології КМ є поняття топології фізичних зв'язків, куди у якості вузлів включають ще й з'єднувальне обладнання, наприклад, комутатори або концентратори. При цьому територіальне розміщення також не враховується. Головною ознакою, за якою легко відрізнити вузол КМ від з'єднувального обладнання, є наявність адреси, без котрої вузол не здатен відправляти чи приймати дані.
3. Існують три варіанти топології окремих (не поєднаних між собою) КМ – шинна (спільна шина), кільцева ("кільце") та зіркоподібна ("зірка"). Прикладом шинної топології є мережі *ALOHA* (експеримент 1971 р.) та *Ethernet* (з 1984 року по наш час), кільцевої – *Token Ring* (1984-2004 роки) та *FDDI* (1987-1994 роки)., зіркоподібної – *ARCNet* (1983-1990 роки).
4. Шинна топологія, яка була прийнята за основу технології *Ethernet,* не зважаючи на можливість колізій (зіткнення і втрата пакетів даних у спільній шині), виявилась найбільш придатною для побудови КМ.
5. У складних мережах, які є об'єднанням будь-якої кількості окремих мереж визначають два типи вузлів: хости (*hosts*), які відправляють та приймають дані, і маршрутизатори (*routers*), які пересилають пакети даних з однієї мережі до іншої.
6. Топологія фізичних зв'язків мережі *Ethernet* може бути зіркоподібною або деревоподібною, але не кільцевою.

7. У складних мережах для з'єднання маршрутизаторів між собою може використовуватись будь-який варіант топології, включаючи кільця.
8. Масштаб КМ (на мові оригіналу *geographic scale*) означає розмір території, на якій розміщено мережу..
9. На початку створення КМ їх за масштабом розподіляли на локальні (***Local Area Network, LAN***) та глобальні (*Wide Area Network, WAN*).
10. Глобальні мережі територіально не обмежені і їх розподіляють на дві категорії: **магістральні** (*Backbone*), що забезпечують зв'язок між містами, країнами, континентами, та **мережі доступу** (*Access network*), які забезпечують зв'язок між віддаленою мережею або віддаленим комп'ютером з деякою головною мережею.
11. Проміжне місце між локальними та глобальними мережами займають **регіональні мережі** (*Metropolitan Area Network, MAN*), що обслуговують територію міста, та **кампусні мережі** (*Campus Area Network, CAN* від англ. *campus* – університетське містечко).
12. Для мереж мінімального масштабу у 2015 році було прийнято стандарт під назвою *Nanoscale and Molecular Communication Framework* (нанорозмірна та молекулярна комунікація). Таким мережам надали назву ***Nanoscale network*** (нанорозмірна мережа**)**.
13. Протягом останніх років з'явилось чимало нових позначень масштабу мереж, наприклад, ***Personal area network, PAN***. (Персональна мережа), ***Body area network, BAN*** (Мережа людського тіла), ***Home area network, HAN*** (Домашня мережа).
14. У галузі КМ поняття **протокол** або телекомунікаційний протокол, можна вважати базовим. Воно означає правила обміну даними між процесами, що відбуваються на різних вузлах мережі.
15. Множину протоколів, якої достатньо для створення мережі, називають **стек протоколів**.
16. Протягом останніх 30 років найбільшого розповсюдження набув **стек *TCP/IP***, на якому побудована мережа Інтернет. Розробники цього стеку розподіляли протоколи на чотири рівні: прикладний, транспортний, міжмережевий та мережевих інтерфейсів.
17. До появи на ринку стеку *TCP/IP* було стандартизовано у 1980 році стек ISO/OSI, у якому протоколи розподілялись на 7 рівнів. Цей стек ніде не використовують з 1990-х років. Протоколи, які були створені для цього стеку втратили актуальність і не підтримуються.

18. Згідно стандарту *ISO/OSI* кожен з семи рівнів є обов'язковим, що не дозволяє відкидати непотрібні рівні. Користь, що залишилось у спадок від стеку *ISO/OSI*, це загальний принцип розподілу протоколів на ієрархічні рівні.
19. Сучасні розробники мереж вважають, що стандартизувати треба лише протоколи, а формування стеків не потребує стандартизації. Це надає можливість формувати стеки з існуючих протоколів для потрібних застосувань без прив'язки до наперед визначеної кількості рівнів.
20. Функціонування програмного забезпечення на кожному з рівнів стеку може відбуватись за різними протоколами та у різних режимах. Важливо, щоб у кожному з'єднанні між відправником і одержувачем даних на усіх рівнях стеку було досягнуто узгоджену взаємодію.
21. Технології фізичного і канального рівнів стандартизуються Інститутом інженерів з електротехніки та електроніки *IEEE* (*Institute of Electrical and Electronics Engineers*), а саме групою *IEEE* 802. Це стандарти для кабельних *Ethernet* мереж (*IEEE* 802.3), а також для *WiFi* бездротових (*IEEE* 802.11) і для *WiMAX* бездротових (*IEEE* 802.16) мереж. Усі ці стандарти об'єднують фізичний і канальний рівні.
22. У нових стандартах фізичний і канальний рівні складаються з низки підрівнів. Підрівні фізичного та канального рівнів являють собою спеціалізовані протоколи, які дозволяють пристроям домовлятись між собою щодо вибору найкращого режиму роботи в умовах появи чи зміни фізичного з’єднання.
23. Розвиток технологій канального рівня має тенденцію до об'єднання усіх раніше стандартизованих технологій так, щоб із новою технологією забезпечувалась можливість використання старих стандартів. Це зручно для користувачів, бо позбавляє від проблем у разі поновлення засобів доступу до мережі.
24. Узгодження взаємодії протоколів стеку *TCP*/*IP* відбувається завдяки тому, що у всіх заголовках пакетів є номер, який вказує на наступний протокол вищого рівня. На канальному рівні це поле *Type*, де у більшості випадків вказується число 2054, що означає *IP* протокол. У *IP* заголовку це поле, де вказано номер протоколу вищого рівня, а у пакетах транспортного рівня це номер порту одержувача.
25. На прикладний рівень потрапляє повідомлення чи потік даних із заголовком, де вказано ім'я та місце знаходження потрібного файлу або команда, яку повинен виконати сервер. Це повідомлення або потік даних

має таку ж послідовність байт, яка була на вході прикладного рівня відправника.

Запитання та завдання для самоперевірки

1. Чим відрізняється поняття топології комп'ютерної мережі (КМ) від топології фізичних зв'язків?
2. Наведіть приклади топології окремих (не поєднаних між собою) комп’ютерних мереж.
3. Якою може бути топологія фізичних зв'язків у мережі *Ethernet*?
4. Яка ознака відрізняє вузол КМ від з'єднувального обладнання?
5. Яка роль маркера у мережах з різною топологією?
6. Яка топологія окремих КМ виявилась найбільш ефективною і чому?
7. Назвіть типи вузлів у мережі Інтернет і надайте їх характеристики.
8. Чи є масштаб КМ територіальною ознакою?
9. Як розподіляють КМ за масштабом?
10. Чим відрізняються магістральні мережі від мереж доступу?
11. Наведіть приклади мереж різного масштабу.
12. Розкрийте поняття телекомунікаційного протоколу та стеку протоколів.
13. Які рівні протоколів є у стеку *TCP/IP* та яке їх призначення?
14. Що являє собою стек *ISO/OSI* і які перспективи його розвитку?
15. Наведіть приклади сучасних стеків протоколів.
16. Яку роль відіграють *RFC* у розвитку Інтернету?
17. Які організації приймають участь у підтримці та розвитку Інтернету?
18. Чим забезпечується узгоджена взаємодія протоколів?
19. Яку роль відіграють підрівні фізичного та канального рівнів?

Р О З Д І Л 2

КАНАЛИ ЗВ'ЯЗКУ КОМП'ЮТЕРНИХ МЕРЕЖ

2.1. Фізичні принципи побудови каналів зв'язку

2.1.1. Основні поняття та термінологія

Кожний канал зв'язку являє собою сукупність фізичного середовища для передавання сигналів, а також передавача і приймача сигналів. Під сигналом розуміємо фізичний процес, який використовується для передачі інформації. Сукупність передавача і приймача сигналів прийнято називати апаратурою для утворення каналу. Канал зв'язку є складовою частиною системи передачі інформації, схему якої зображено на рис.2.1.

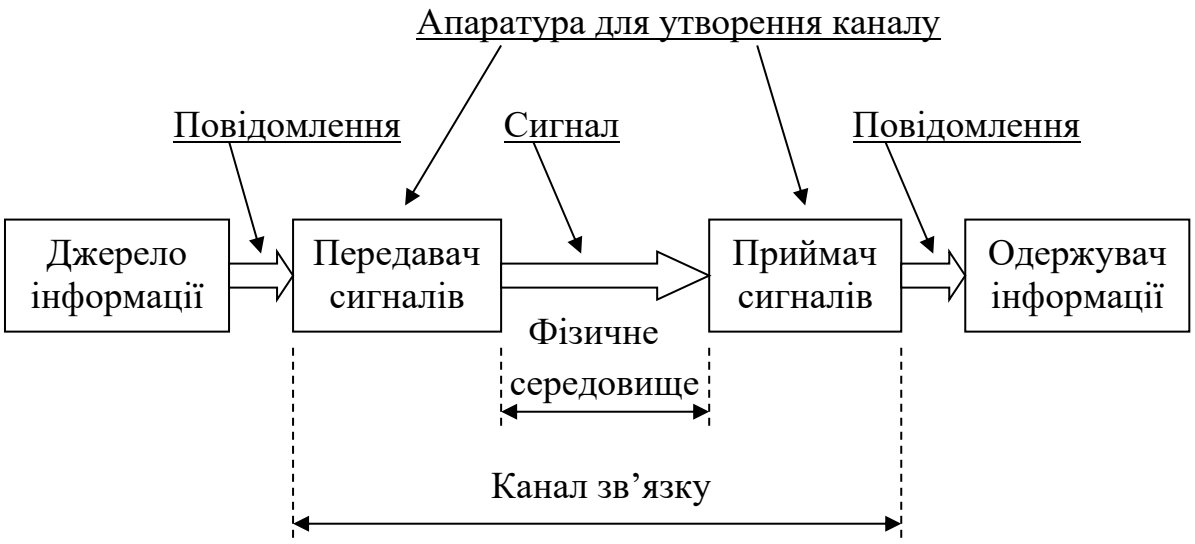


Рис.2.1. Канонічна схема системи передачі інформації

У комп'ютерних мережах інформацію прийнято називати даними, тому канали зв'язку в цих мережах називають каналами передачі даних.

У сучасних комп'ютерах вся інформація зберігається і обробляється у вигляді послідовностей двійкових одиниць (*binary digit*, від цих слів походить скорочення *bit* – біт), які позначають цифрами 0 та 1. Цю форму представлення інформації називають дискретною або цифровою. Сигнали, якими передають дискретну інформацію, також називають дискретними. Такі сигнали мають обмежену кількість варіантів.

2.1.2. Аналогово-дискретні та дискретно-аналогові перетворення

Процеси, які ми спостерігаємо у реальному житті, у більшості мають безперервну форму і для комп'ютерної обробки їх необхідно перетворити у дискретний вигляд, а після комп'ютерної обробки можливо потрібно буде надати результату безперервний вигляд. Такі перетворення називають аналогово-дискретними та дискретно-аналоговими відповідно. Розглянемо процедуру цих перетворень на прикладі телефонії.

Звуки, які ми пересилаємо телефоном, являють собою безперервні процеси. Після перетворення звуків за допомогою мікрофону в електричні коливання отримуємо теж безперервний процес, ділянку якого зображено у вигляді графіку на рис.2.2.

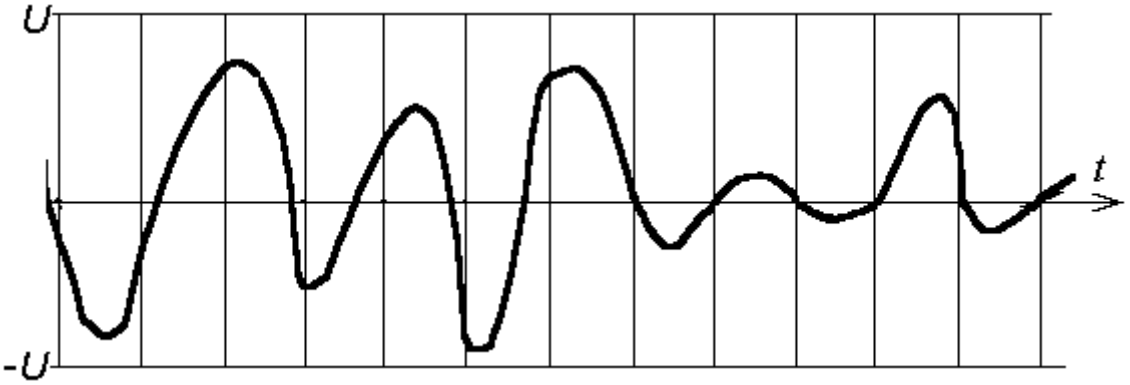


Рис.2.2. Коливання електричної напруги на виході мікрофону

Процедура перетворення безперервного процесу у дискретний має назву дискретизація (*sampling*). Ця процедура полягає в тому, що через деякі рівні проміжки часу, які на рисунку показано у вигляді вертикальних ліній, вимірюють цифровим вольтметром електричну напругу. Перед цим необхідно задати інтервал [-*U*, *U*] можливих значень напруги, яку вимірюють. Результат кожного вимірювання називають відліком [3].

Важливою задачею дискретизації є вибір оптимальної частоти вимірів. Цю задачу було розв'язано відомим вченим В.О. Котельниковим [4].

Теорема Котельникова (її також називають теоремою Найквиста-Шеннона) проголошує, що процес зі обмеженим спектром можна без втрат відтворити у разі коли частота вимірів більше ніж у двічі перевищує максимальну частоту спектру цього процесу.

Нагадаємо, що спектр процесу *s(t)* являє собою пряме перетворення Фур'є, яке можна описати формулою

$$S(j\omega) = \int_{-\infty}^{\infty} s(t) e^{-j\omega t}\, dt, \qquad (2.1)$$

де $j = \sqrt{-1}$; $\omega = 2\pi f$;

f – частота в герцах.

Всі ці математичні дослідження виконані в умовах нескінченого інтервалу часу, але їх результати з успіхом використовують у реальних задачах. Для практичних розрахунків замість інтегралу обчислюють суму елементів ряду Фур'є, припускаючи, що досліджуваний процес є періодичним з періодом *T*. При цьому безперервний спектр стає дискретним і являє собою ряд синусоїд зі частотами, що кратні значенню 1/*T*. Синусоїди, на які можна розкласти сигнал, називають гармонійними складовими або гармоніками. Для досягнення необхідної точності розрахунків обирають достатньо велике значення періоду *T*. Приклад такого дослідження проілюстровано у додатку (див. дод. 4). Фактично за допомогою перетворення (2.1) який завгодно фізичний процес може бути представлений як сума гармонік.

У первісних системах телефонії звук у вигляді електричних коливань (див. рис. 2.2) і був сигналом, який безпосередньо пересилався по дротових лініях зв'язку у вигляді електричного струму. Дослідження виявили, що у лініях зв'язку різні гармоніки ослаблюються по-різному, а з підвищенням частоти сигнали втрачаються. При цьому тільки гармоніки з частотами від 300 до 3400 Гц в значній мірі впливають на якість звуку, а ослаблення інших гармонік не має суттєвого значення. Тому діапазон 300-3400 Гц прийнято за стандарт у телефонії. Звідси витікає, що спектр сигналів телефонії можна вважати обмеженим на частоті 3400 Гц.

Для дискретизації сигналів телефонії згідно з розглянутою теоремою було обрано частоту вимірів 8 кГц, а значення електричної напруги для кожного виміру перетворюють у 8-ми бітове число, тобто у один байт. Це означає, що похибка від заміни фактичного значення електричної напруги на числове не перевищуватиме 0,5 % від обраної максимальної величини *U*. Перелічені параметри увійшли в міжнародний стандарт цифрової телефонії. Таким чином замість аналогових сигналів у цифровій телефонії пересилають потік бітів зі швидкістю 64 кбіт/с, що дозволяє для передачі телефонних розмов використовувати канали комп'ютерних мереж.

2.1.3. Імпульсні процеси та їх спектри

По каналах зв'язку комп'ютерних мереж пересилаються виключно дискретні сигнали. Найбільш вдалою формою таких сигналів вважають імпульсні процеси, які прийнято називати імпульсами. Під імпульсними розуміють процеси, тривалість яких сумірна з тривалістю нестаціонарних процесів у фізичному середовищі, де вони відбуваються. Зрозуміло, що для збільшення швидкості передачі інформації слід зменшувати тривалість імпульсів, але через нестаціонарні процеси фронти (передній і задній краї) імпульсів стають похилими (розтягуються у часі), через що імпульси накладаються один на одного і їх стає важко розпізнавати.

Користуючись формулою (2.1) знайдемо спектр для прямокутного імпульсу, який зображено на рис.2.3.

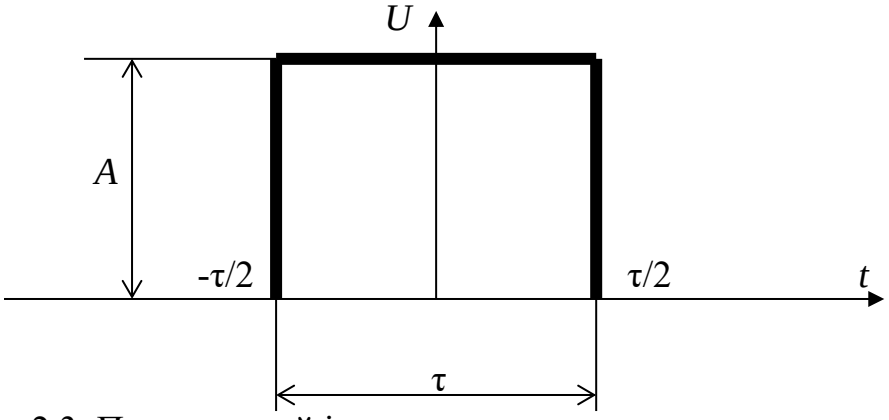


Рис.2.3. Прямокутний імпульс для розрахунку спектру

Для спрощення математичних перетворень будемо вважати, що наш імпульс починається в момент -τ/2 та закінчується в момент τ/2. Відомо, що на спектр сигналів не впливає їх переміщення в часі. Зробивши заміну параметрів у формулі (2.1) на ті, що відповідають нашому імпульсу, отримуємо наступний вираз для спектру цього імпульсу.

$$S_{imp}(j\omega)=\int_{-\tau/2}^{\tau/2} Ae^{-j\omega t}dt=-\frac{A}{j\omega}\int_{-\tau/2}^{\tau/2} e^{-j\omega t}d(-j\omega t)$$

Підставляючи межі інтегрування отримуємо наступний результат.

$$-\frac{A}{j\omega}\left(e^{-j\omega\frac{\tau}{2}}-e^{j\omega\frac{\tau}{2}}\right)=\frac{2A}{\omega}\left(\frac{e^{j\omega\frac{\tau}{2}}-e^{-j\omega\frac{\tau}{2}}}{2j}\right)$$

Останній вираз у дужках являє собою формулу Ейлера для функції синусу. Скориставшись цією формулою та заміною ω на 2π*f*, отримаємо остаточний результат.

$$\frac{2A}{\omega}\sin\frac{\omega\tau}{2}=\frac{A\tau}{\pi\tau f}\sin(\pi\tau f)$$

Графічне зображення спектру амплітуд, що побудовано за останнім виразом для абсолютних значень функції sin, показано на рис. 2.4.

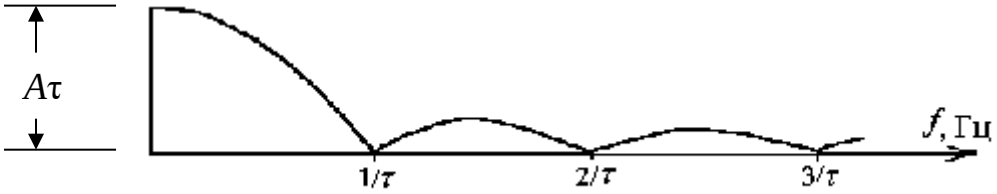


Рис.2.4. Спектр амплітуд для прямокутного імпульсу

Як бачимо з побудованого графіку, при умові наближення значення частоти *f* до нуля спектр наближається до розміру площі нашого імпульсу.

Відомо, що спектр усіх процесів, які мають обмежену тривалість, буде нескінченим. Немає реальних каналів зв'язку, які б мали нескінчену смугу частот передавання, тому неможливо у повній мірі позбутися викривлень імпульсів під час передавання.

Ділянки спектру в інтервалах довжиною 1/τ прийнято називати пелюстками. Частка енергії, яка зосереджена у першому (головному) пелюстку становить близько 90%, що вважають достатнім для якісного зв'язку. Тому для передавання імпульсів тривалістю τ обирають смугу частот не більшу за 1/τ .

За допомогою зворотного перетворення Фур'є можна знайти форму сигналів виходячи зі спектру. Цю задачу для випадку спектру, що має прямокутну форму з максимальною частотою *F*, розв'язано у роботі [4]. Результат у вигляді процесу *S*(*t*) описує формула

$$S(t)=\frac{1}{2\pi Ft}\sin 2\pi Ft. \qquad (2.2)$$

Графік, який побудовано за формулою (2.2), надано на рис. 2.5.

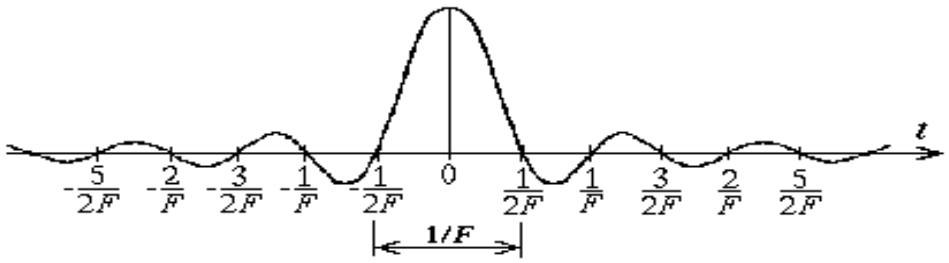


Рис.2.5. Форма сигналу, що має рівномірний спектр від 0 до *F*.

2.1.4. Властивості спектральних характеристик процесів

Як бачимо з виразу (2.2), процес $S(t)$ є нескінченим і являє собою добуток гіперболи зі синусоїдою. Процес такої форми називають функцією або імпульсом Котельникова. Особливість цього процесу полягає в тому, що його форма не повторюється ні на яких двох ділянках. Достатньо виміряти його значення в якому завгодно інтервалі у минулому або у майбутньому часі для відтворення усього процесу. Це означає, що процеси з обмеженим спектром мають властивість передбачуваності. Цей момент не може не зацікавити допитливих дослідників, бо в ньому приховується парадокс. Ми вважаємо, що багато реальних процесів мають початок і кінець, а з математичної точки зору такі процеси повинні мати нескінчений спектр. Відомо, що максимальна швидкість руху будь-яких об'єктів не може перевищувати швидкість світла, а при умові обмеження швидкості навіть мінімальна частка матерії не може коливатись із нескінченою частотою. Це обумовлює обмеження спектру, а всі процеси з обмеженим спектром є нескінчені і не можуть мати ні початку ні кінця, крім того їх можна передбачати. Але, чи існує мінімальна частка матерії? Від того як почали ділити атом, який вважали мінімальною часткою, складається враження, що кожну частку можна поділити на ще менші і цей процес нескінчений. За таких умов частота коливань також може зростати нескінченно. З іншого боку, ми бачимо, що рух великих часток матерії таких як планети, можна передбачати з високою точністю.

На основі аналізу функції $S(t)$ одержано два важливих висновки.

По-перше, якщо пересилати сигнали такої форми з інтервалом $1/2F$, то їх накладання один на одного не буде заважати вимірюванню амплітуди кожного з них, бо в ці моменти сума всіх інших сигналів дорівнює нулю. Звідси витікає, що максимальна частота передавання імпульсних сигналів не може перевищувати $2F$. Це розглянуто у додатку (див.дот. 4).

По-друге, для того, щоб з абсолютною точністю відтворити процес з обмеженим спектром по вимірам з частотою $2F$, де F – максимальна частота спектру, треба просумувати послідовність функцій $S(t)$ у яких амплітуди дорівнюють значенням відповідних вимірів.

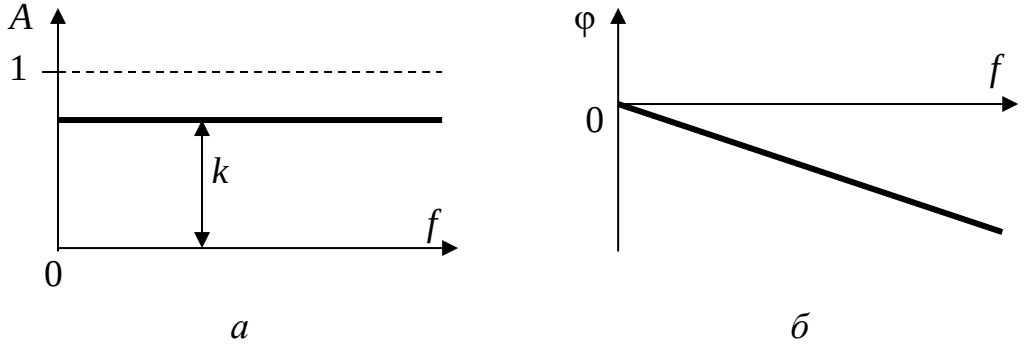
Оскільки неможливо формувати нескінченно довгі сигнали, можна обмежити їх довжину, виходячи з вимог до точності формування спектру. Пристрій для формування таких сигналів запропоновано В.М.Вишняковим [5]. Переваги від використання сигналів, що схожі за формою на функцію

Котельникова полягають у тому, що їх форма мало змінюється під час передавання і вони не створюють завад за межами своєї смуги частот.

2.1.5. Зміни сигналів у середовищах передавання

На сигнали під час передавання впливають детерміновані та випадкові фактори.

До детермінованих відносять затримку, яка обумовлена швидкістю розповсюдження енергії у фізичному середовищі. Ця затримка чітко пов'язана із відстанню передавання і позбутися її неможливо, бо в сучасних каналах зв'язку швидкість передавання сигналів дуже близька до швидкості світла. Також детермінованим вважають перетворення форми сигналу через відхилення амплітудно-частотної (АЧХ) та фазово-частотної (ФЧХ) характеристик середовища передавання від ідеальних, які зображено на рис. 2.6.



а *б*

Рис.2.6. Ідеальні варіанти АЧХ (*а*) та ФЧХ (*б*) фізичного середовища:

A – амплітуда (на вході в середовище дорівнює одиниці); φ – фаза; *f* – частота; *k* – коефіцієнт передавання амплітуди (0< *k*<1).

За умов ідеальних АЧХ та ФЧХ сигнали можуть ослаблюватись та затримуватись, а їх форма залишиться незмінною.

Ослаблення залежить від значення *k*, а затримка пропорційна тангенсу кута нахилу ФЧХ.

У реальних фізичних середовищах не буває ідеальних АЧХ та ФЧХ. Для таких середовищ введено поняття смуги частот перепускання. Це той діапазон частот, у якому слід вибирати значення робочих частот для формування сигналів. Вважають, що у цьому діапазоні АЧХ та ФЧХ у певній мірі наближені до ідеальних. Форма сигналів, що проходять крізь реальне середовище передавання, завжди змінюється. Ці зміни можна визначити за допомогою математичних методів після вимірювання АЧХ та ФЧХ.

Зміну форми сигналів, що відбувається у середовищі передавання через відхилення АЧХ та ФЧХ від ідеальних, називають спотворенням.

Випадкові фактори, які впливають на сигнали у каналах зв'язку, називають завадами.

За типом впливу на сигнал завади розподіляють на дві категорії:

- адитивні – це випадкові процеси, що додаються до сигналів;
- мультиплікативні – це випадкові зміни коефіцієнта *k*.

Завади являють собою випадкові процеси, параметри яких описують за допомогою методів теорії ймовірностей.

2.1.6. Оптимізація систем передачі інформації

В кожному каналі зв'язку існує рух молекул, що утворює заваду, яку називають адитивний білий Гаусів шум. Структурна схема оптимального приймача сигналів в умовах такого шуму, яка запропонована в роботі [4], представлена на рис. 2.7.

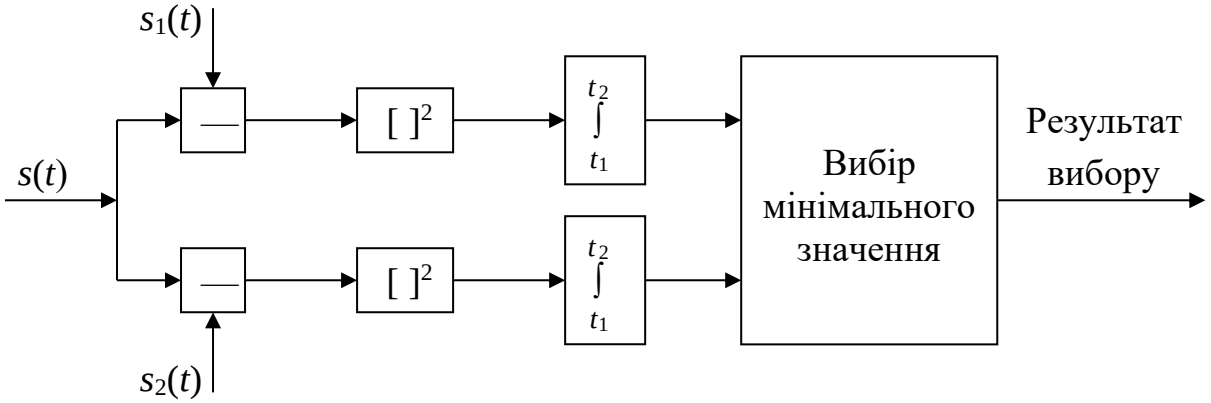


Рис.2.7. Схема оптимального приймача В.А.Котельникова.

Такий приймач забезпечує мінімум помилок при умові, що сигнал $s(t)$ має два можливих варіанти $s_1(t)$ або $s_2(t)$ у інтервалі часу $[t_1, t_2]$.

Для довільної кількості варіантів сигналу $s(t)$ оптимальний приймач повинен обирати варіант з номером i, який відповідає мінімальному значенню виразу

$$\int_{t_1}^{t_2} \left[s(t) - s_i(t)\right]^2 dt, \qquad (2.3)$$

де $i = 1, 2, \ldots n$, n – кількість варіантів сигналу.

Після розкриття квадрату у виразі (2.3) отримуємо наступну суму інтегралів.

$$\int_{t_1}^{t_2}\left[s^2(t)-2s(t)\cdot s_i(t)+s_i^2(t)\right]dt=\int_{t_1}^{t_2}s^2(t)dt-2\int_{t_1}^{t_2}s(t)\cdot s_i(t)dt+\int_{t_1}^{t_2}s_i^2(t)dt\,.$$

Для випадку коли усі варіанти сигналів мають однакову енергію, перший і останній доданки не впливають на результат порівняння. При цьому мінімальному значенню виразу (2.3) буде відповідати максимальне значення виразу

$$\int_{t_1}^{t_2}s(t)\cdot s_i(t)dt\,. \qquad (2.4)$$

Схема оптимального приймача, який може бути побудований на основі виразу (2.4), спрощується у порівнянні із попередньою схемою (див. рис. 2.7).

Віднімання можливих варіантів сигналу із процесу $s(t)$ у попередній схемі ускладнює можливість побудови приймача в реальних умовах, бо необхідно не тільки з високою точністю визначити моменти t_1 та t_2, але й точно враховувати ослаблення сигналу.

Спрощений на основі виразу (2.4) варіант схеми оптимального приймача, який позбавлено від цього недоліку, зображено на рис. 2.8.

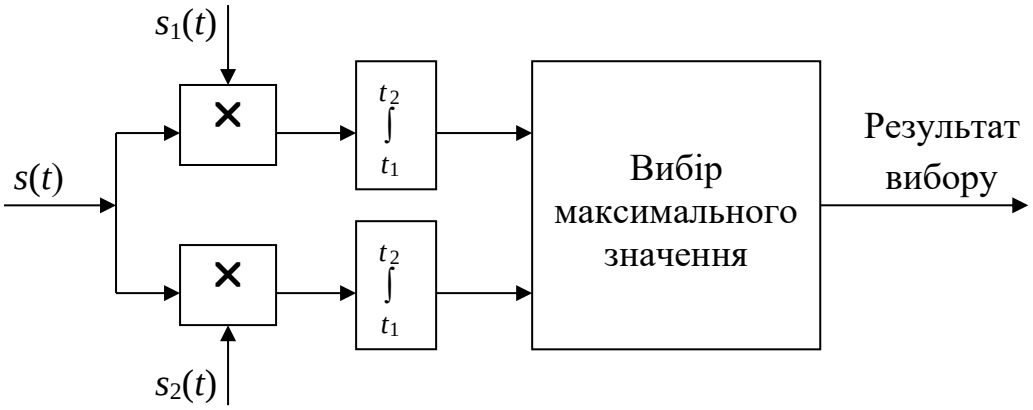


Рис.2.8. Спрощений варіант схеми оптимального приймача.

Розробка схеми оптимального приймача допомогла сформулювати поняття відстані між сигналами та оптимальної множини сигналів. Для найпростішого випадку, а саме у разі двох варіантів сигналу, оптимальною є пара протилежних один одному сигналів. При цьому відстань, яка вимірюється кількістю енергії, що необхідна для перетворення одного сигналу в процес більш схожий на другий сигнал, є максимальною. Для систем, у яких $n>2$, оптимальною є множина ортогональних сигналів [6].

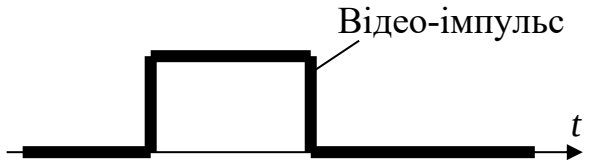
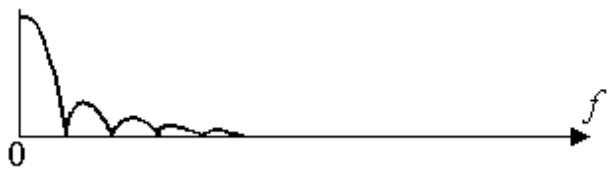
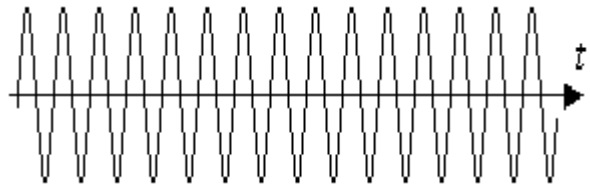
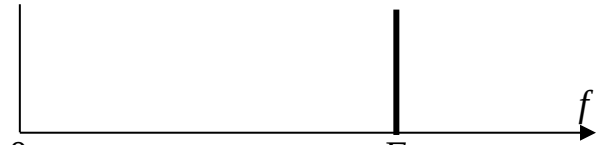
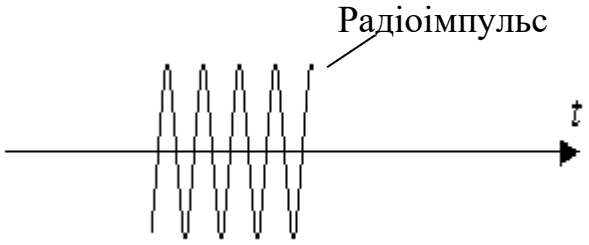
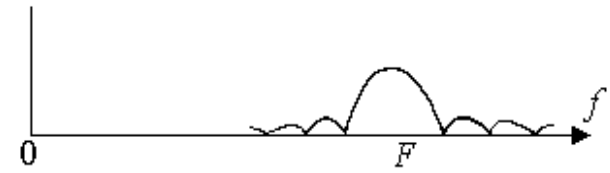
2.1.7. Спектральні перетворення сигналів і модуляція

У багатьох каналах зв'язку смуга частот передавання не містить низькочастотних складових. В першу чергу це стосується всіх систем радіозв'язку. У цих системах неможливо пересилати імпульси, спектр яких починається зі частоти, що наближається до нуля, і виникає необхідність у спектральному перетворенні сигналів, яке називають модуляцією.

Найпростіший варіант модуляції можна представити у вигляді процедури множення імпульсу, який треба передати, на синусоїду, що має частоту у потрібному діапазоні. Принцип роботи модулятора для цього варіанту проілюстровано графіками у таблиці 2.1.

Таблиця 2.1

Вигляд процесів та їх спектрів під час модуляції

Форма процесу у часі	Спектр процесу
	
 *F* – частота коливань	
	

Імпульс (на вході модулятора), який містить у своєму спектрі низькочастотні складові прийнято називати відео-імпульсом.

Синусоїдальний процес зі частотою F називають носієм, а частоту F – частотою носія або несучою частотою.

Сигнал, який отримують в результаті спектрального перетворення, називають радіоімпульсом.

За допомогою модуляції спектр сигналу переносять у потрібний частотний діапазон для забезпечення можливості передавання крізь задане фізичне

середовище. Після передавання радіоімпульс перетворюють у відео-імпульс за допомогою демодулятора.

Як бачимо з графіків (див. табл. 2.1), спектр радіоімпульсу у два рази ширший за спектр відео-імпульсу і має вісь симетрії на частоті *F*. Це пояснюється тим, що ми не розглядаємо від'ємні значення частоти, а додаємо амплітуди гармонік від'ємної частоти до відповідних гармонік додатної частоти. Якщо б ми скористались поняттям від'ємної частоти, то спектр відео-імпульсу прийняв таку ж форму, як для радіоімпульсу, з віссю симетрії на нульовій частоті. На наших графіках спектр відео-імпульсу у два рази вищій за спектр радіоімпульсу, що відповідає умові однакової енергії відео та радіо імпульсів.

У загальному випадку під модуляцією розуміють процедуру зміни параметру високочастотного процесу під впливом низькочастотного процесу, що несе інформацію.

Для імпульсних сигналів використовують три основних типи модуляції, а саме амплітудну, частотну та фазову. Розглядаючи відрізок синусоїди в межах радіоімпульсу легко пояснити ці типи модуляції за допомогою формули $A\sin(2\pi F+\varphi)$, яка описує даний відрізок, де A, F та φ – амплітуда, частота та фаза відповідно, що і являють собою ті параметри, які можна змінювати.

В деяких системах застосовують комбіновану модуляцію, яка полягає в одночасній зміні двох параметрів. Так у модемах для телефонних ліній обрано амплітудно-фазову модуляцію зі числом можливих значень комбінацій амплітуди та фази до декількох десятків.

Базуючись на симетрії спектру радіоімпульсу, з метою більш раціонального використання частотного діапазону, створюють системи зі передаванням однієї половини спектру імпульсу, оскільки другу половину можна відтворити на боці приймача за правилом симетрії.

2.2. Фізичний та канальний рівні моделі *ISO/OSI*

2.2.1. Відповідність між стеком *TCP/IP* та моделлю *ISO/OSI*

Розробники стеку *TCP/IP* не розглядали окремо фізичний та канальний рівні. На цьому місці в їхній ієрархії рівнів бачимо тільки один рівень мережевого інтерфейсу. Ось як це можна пояснити. Задача, яка тоді вирішувалась, полягала у створенні протоколів вищих рівнів, а до нижніх рівнів, які вже існували в різних мережах, необхідно було тільки приєднатись, тобто розробити до них інтерфейси.

Задача створення міжнародного стандарту побудови комп'ютерних мереж повинна була охоплювати усі процедури, які відбуваються між прикладними процесами серверів та клієнтів, включаючи апаратні засоби, які приймають участь у реалізації цих процедур. Рівні моделі *ISO/OSI* були запропоновані виходячи з можливості незалежного розв'язання задач на кожному з цих рівнів.

Розподіл задачі побудови каналу зв'язку для комп'ютерної мережі на фізичний та канальний рівні у моделі *ISO/OSI* є досить чітко визначеним.

Фізичний рівень охоплює усі фізичні характеристики пристроїв, які необхідні для приєднання до фізичного середовища, а крім того характеристики фізичних процесів, які використовують для передавання інформації. На цьому рівні визначаються форма, розміри та інші фізичні параметри роз'єднувачів та антен, а також потужність, частота та інші фізичні характеристики сигналів.

На канальному рівні обумовлюються правила формування пакетів канального рівня (такі пакети прийнято називати кадрами) у вигляді послідовності біт. Це формати заголовків та кінцівок кадрів і обмеження, що накладаються на кількість і порядок розміщення даних в кадрі. Також на цьому рівні визначаються алгоритми передавання кадрів.

Більш глибокий розгляд механізмів фізичного та канального рівня дозволяє кожен з цих рівнів розподілити на підрівні.

У фізичному рівні можна виділити підрівень, у якому обумовлюють форму та інші фізичні параметри роз'єднувачів, а характеристики сигналів віднести до іншого підрівня. Розподіл тут достатньо чіткий, але у технологіях фізичного рівня існує деяка залежність між роз'єднувачами та сигналами. Наприклад, оптичні сигнали до волоконно-оптичного кабелю потрапляють через оптичний роз'єднувач, а для електричних сигналів потрібен електричний роз'єднувач.

У деякій апаратурі, що реалізує функції фізичного та канального рівнів, можна нарахувати близько десятка ієрархічних підрівнів. Один з таких варіантів розподілу канального і фізичного рівня на підрівні для технології *Ethernet* зображено на рис. 1.9.

Пояснення щодо виникнення такої ієрархії підрівнів полягають у наступному.

У засобах, що призначені для приєднання до мережі за технологією *Ethernet*, передбачена можливість використання декількох різновидів даної технології. Швидкість передавання для одного з перших, але який ще до цього часу перебуває в експлуатації, варіанту технології *Ethernet* становить 10 Мбіт/с (стандарт 1991 р.), у наступному варіанті *Fast Ethernet* швидкість підвищено до

100 Мбіт/с (стандарт 1995 р.), а у подальшому варіанті *Gigabit Ethernet* швидкість підвищено до 1000 Мбіт/с (стандарт 1998 р.). Крім того, для кожного з перелічених варіантів можуть використовуватись різні типи кабелів і різні методи модуляції. Слід зауважити, що модуляцію іноді називають кодуванням на фізичному рівні (*Physical Coding*). Підрівні фізичного рівня являють собою спеціалізовані протоколи, які дозволяють пристроям автоматично домовлятись між собою з метою вибору найбільш ефективного режиму роботи в умовах кожного фізичного з'єднання.

Підрівнями канального рівня керують протоколи вищих рівнів.

Інтерфейс між канальним і третім за ієрархією мережевим рівнем моделі *ISO/OSI* точно відповідає аналогічному інтерфейсу стеку *TCP/IP*, але для рівнів, які розміщені вище третього, точної відповідності ми не спостерігаємо.

2.2.2. Характеристики обладнання фізичного рівня

Головна задача фізичного рівня полягає в забезпеченні надійного інтерфейсу з фізичним середовищем. Це, з одного боку, створення засобів апаратного приєднання до середовища, а, з другого боку, вибір сигналів, які б забезпечили потрібну швидкість передавання інформації.

Для передачі даних використовують штучні та природні середовища.

До штучних слід віднести електричні та оптичні кабелі зв'язку, а до природних – радіо-ефір та вільний простір, крізь який пересилають сигнали у вигляді променів інфрачервоного світла. У деяких випадках для передачі даних використовують штучні середовища іншого призначення, наприклад, кабелі енергопостачання або трубопроводи, але такі рішення є винятковими і не знаходять широкого впровадження.

Більше ніж 100 років електричні кабелі зв'язку були головним фізичним середовищем для передавання сигналів на будь-яку відстань. Суттєві зміни відбулися на початку 21-го сторіччя. З одного боку, розвиток мережі Інтернет став причиною швидкого зростання потреб у передаванні великих обсягів інформації на значні відстані, що не в змозі були задовольнити існуючі електричні канали зв'язку. З другого боку, наукові досягнення у створенні волоконно-оптичних кабелів зв'язку забезпечили можливість випереджувати зростаючі потреби у швидкості та надійності передачі інформації на будь яку відстань в межах земної кулі. Як перше, так і друге, стимулювало стрімкість прокладання оптичних ліній зв'язку між континентами та країнами і поступову заміну усіх магістральних кабелів зв'язку з електричних на волоконно-оптичні.

Електричні кабелі поки що мають переваги в мережах для офісів та будинків. Розглянемо характеристики найбільш розповсюджених типів цих кабелів.

В перших варіантах мереж *Ethernet* використовували коаксіальний кабель двох типів: товстий (∅ 9,5 мм) та тонкий (∅ 5 мм). Частота носія становила 10 МГц, а максимальна швидкість передавання – 10 Мбіт/с. Ці варіанти технології *Ethernet* не рекомендовані для використання ще з 90-х років минулого сторіччя. У наступних варіантах технології *Ethernet* коаксіальний кабель було замінено на кабель типу скручена пара, що забезпечило можливість підвищення швидкості передачі даних у 10 і 100 разів та збільшило надійність з'єднань. Єдиною перевагою технологій зі коаксіальним кабелем є більша у порівнянні із кабелем типу скручена пара максимальна довжина з'єднання (без підсилювачів), яка становить 500 м для товстого і 185 м для тонкого коаксіального кабелю проти 100 м для скрученої пари.

Коаксіальний кабель має в середині мідну жилу, вздовж якої розповсюджується електромагнітна хвиля. Шар ізоляції навколо жили обгорнено металевою оболонкою, що приєднується до шини заземлення і захищає сигнали від зовнішніх електромагнітних завад. Пластикова оболонка захищає кабель від механічних пошкоджень (рис. 2.1).

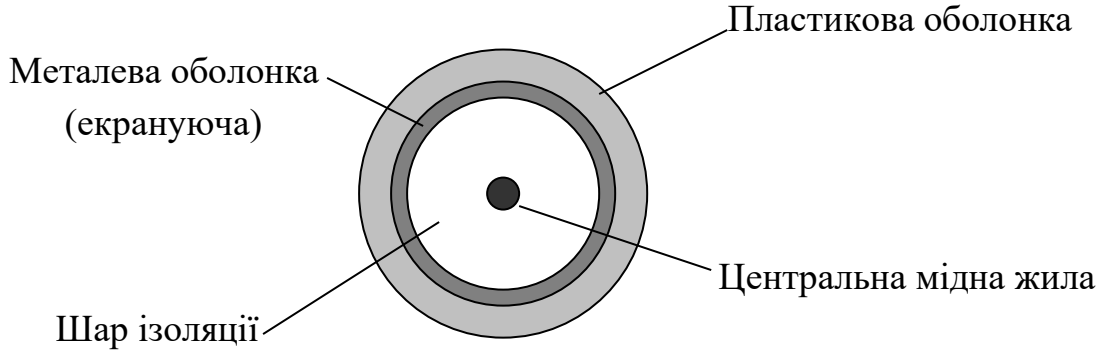


Рис. 2.1. Структура коаксіального кабелю

Фрагмент мережі зі використанням тонкого коаксіального кабелю зображено на рис. 2.2.

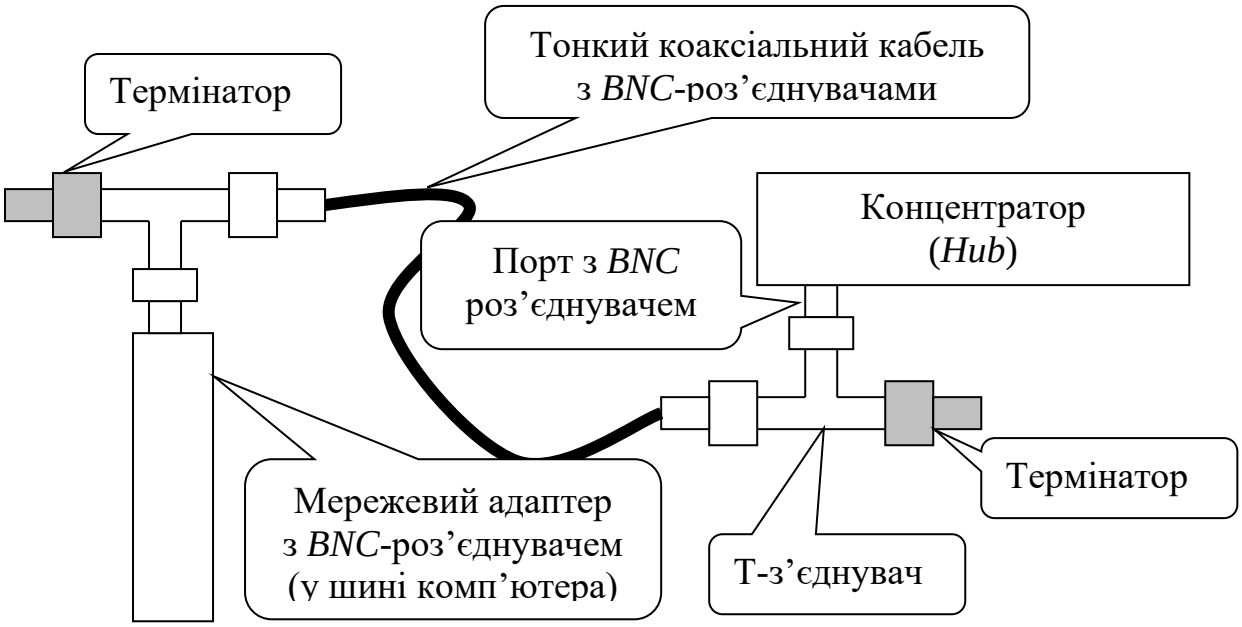


Рис. 2.2. Приєднання комп'ютера до концентратора за допомогою тонкого коаксіального кабелю

Зовнішній вигляд концентратора з елементами, що необхідні для підключення тонкого коаксіального кабелю показано на рис. 2.3.

Рис. 2.3. Концентратор (*HUB*) з *BNC*-роз'єднувачем (на передньому плані зліва направо розміщені: кінець тонкого коаксіального кабелю зі роз'єднувачем, Т-з'єднувач і термінатор)

Такий концентратор дозволяє поєднати на фізичному рівні в одній мережі наступні три варіанти стандартів технології *Ethernet*.

- 10*BASE*-5 – на товстому коаксіальному кабелі.
- 10*BASE*-2 – на тонкому коаксіальному кабелі.
- 10*BASE-T* – на кабелі типу скручена пара.

Число 10 у позначеннях стандартів означає максимальну швидкість передачі 10 Мбіт/с, слово *BASE* є скороченням від *Baseband*, що означає тракт для передачі сигналів, а кінцева цифра чи буква характеризує тип кабелю (5 – товстий коаксіальний, 2 – тонкий коаксіальний, *T* – скручена пара).

Особливість технологій зі використанням коаксіального кабелю полягає в тому, що на обох кінцях кабелю необхідно встановлювати термінатори. Термінатор перетворює електромагнітну енергію в теплову, що не дає можливості утворенню завад через відбивання електромагнітної хвилі від кінців кабелю. Термінатор являє собою звичайний резистор, опір якого дорівнює хвильовому опору коаксіального кабелю, який у нашому випадку дорівнює 50 Ом.

Тонкий коаксіальний кабель буває доцільно використовувати у разі коли довжина з'єднання перевищує 100 м, а швидкість до 10 Мбіт/с є задовільною.

Для з'єднань довжиною до 100 м кабель типу скручена пара має значні переваги не тільки у швидкості обміну інформацією, але також у надійності і простоті обслуговування мережі.

Найбільшою популярністю для побудови ЛКМ користується кабель типу *UTP* (*Unshielded Twisting Pare* – неекранована скручена пара), бо він є найдешевшим серед кабелів для комп'ютерних мереж. Не зважаючи на відсутність екрануючої оболонки, цей кабель досить добре захищений від впливу зовнішніх електромагнітних полів, бо захист забезпечується саме тим, що проводи кожної пари скручують між собою. Зараз пояснимо це докладніше.

Сигнали для передавання по парі проводів формуються симетрично таким чином, щоб полярність імпульсів в проводах пари була протилежна. Такий метод формування сигналів називають диференціальним (рис. 2.4).

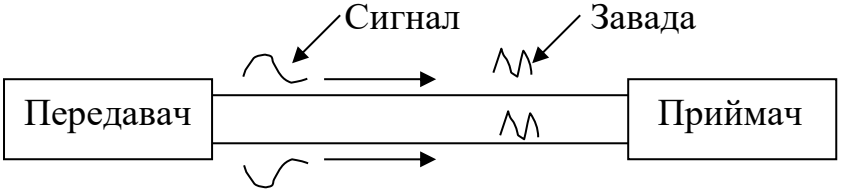


Рис. 2.4. Процес передавання сигналу по парі проводів

Приймач сигналу вимірює різницю потенціалів між проводами пари. Іншими словами він віднімає один від одного процеси, що відбуваються в проводах пари. Оскільки вплив завади на кожен із проводів пари буде однаковим, то енергія завади на вході приймача буде знищуватись. Для того, щоб ці умови виконувались найкращим чином, треба було б обидва проводи пари прокласти в одному й тому ж місці, але це неможливо. Коли проводи пари розміщували поруч

один з одним, як в телефонних кабелях типу ТРП, то дії завад з різних боків впливали по-різному на кожен зі проводів. Щоб забезпечити однакові умови для проводів пари вирішили скручувати їх один з одним. При цьому вдалося не тільки зменшити вплив зовнішніх полів на свій сигнал, але й також зменшити енергію полів від своїх сигналів, бо поля, що утворюються від імпульсів протилежної полярності компенсують одне одного. Останнє дуже важливо коли декілька пар знаходяться в одній кабельній оболонці і можуть своїм полем утворювати завади для сусідніх пар.

Неекрановану скручену пару можна з успіхом використовувати у випадку відсутності потужних електромагнітних завад і коли немає вимог щодо захисту інформації від прослуховування. У випадках наявності таких вимог можна застосовувати екрановану скручену пару *STP* (*Shielded Twisted Pair*), яка має металеву обгортку. Для зовнішніх з'єднань виробляють кабель з міцною пластиковою оболонкою. До неї може бути приєднано сталевий трос для натягування кабелю між будинками, але у цьому разі треба звертати увагу на захист від блискавок. Для найбільш повного захисту від блискавок і від прослуховування можна скористатись волоконно-оптичним кабелем.

Виготовляють одножильну і багатожильну скручену пару. У першої проводи суцільні (мідні або з мідним покриттям), а у другої кожний провід складається із багатьох тоненьких дротиків (рис. 2.5).

Рис. 2.5. Внутрішня структура багатожильного (з лівого боку) та одножильного (з правого боку) кабелю типу *UTP*. Для наочності з кінцівок проводів зрізано ізоляцію. На кінці багатожильного є роз'єднувач *RJ*-45, а одножильний приєднано до розетки

Одножильна скручена пара призначена для прокладання у коробах або закріплення на стінах. На її кінцях встановлюють розетки.

Багатожильна скручена пара призначена для виготовлення гнучких з'єднувальних шнурів (див. рис. 1.4), які мають назву *Patch cord*. На кінцях

шнурів встановлюють вилки 8*P*8*C*, де букви *P* та *C* означають *Position* та *Contact*. Ці шнури можуть бути симетричними (проводи на обох роз'єднувачах розміщені однаково) або асиметричними (проводи розміщені по різним варіантам стандарту) (див. дод. 2).

Симетричні шнури призначені для підключення комп'ютерів до концентраторів або комутаторів та для з'єднання комутаторів між собою.

Асиметричні шнури призначені для з'єднання комп'ютерів між собою та для підключення комп'ютерів до деяких типів модемів.

В місцях розміщення комутаційного обладнання може виникати необхідність встановлення значної кількості розеток. Тоді замість окремих розеток встановлюють *Patch Panel* (патч-панель), що являє собою компактну групу розеток на одній панелі (рис. 2.6).

Рис. 2.6. *Patch Panel* з гніздами під роз'єднувачі типу 8*P*8*C* (верхнє фото) і її зворотний бік з планками для закріплення кінців кабелю (нижнє фото)

Екранована і неекранована скручені пари мають різний хвильовий опір (150 Ом і 100 Ом – відповідно). Незважаючи на те, що гнізда під екрановані і неекрановані роз'єднувачі абсолютно однакові за розміром, вони можуть бути призначені тільки для якогось одного типу кабелю. Це необхідно приймати до уваги під час придбання мережевого обладнання.

За міжнародним стандартом *ISO*/*IEC* 11801 щодо структурованих кабельних мереж старі позначення екранованих та неекранованих кабелів типу скручена пара замінено на нові. У нових позначеннях додано символ "/", ліворуч від якого вказують наявність та тип екрану навколо кабелю, а праворуч – наявність екрану навколо кожної окремої пари в кабелі. На кабелях можна зустріти як нові, так і старі позначення. Їх відповідність надано у вигляді таблиці 2.2.

Таблиця 2.2

Порівняння старих та нових позначень кабелів

Старе позначення	Нове позначення	Екран навколо кабелю	Екран навколо пар
UTP	*U/UTP*	Немає	Немає
FTP	*F/UTP*	Фольга	Немає
STP	*U/FTP*	Немає	Фольга
S-FTP	*SF/UTP*	Фольга та сітка	Немає

Букви у позначеннях означають таке: *TP – twisted pair* (скручена пара); *U – unshielded* (неекранована); *F – foil shielding* (екранована фольгою); *S – braided shielding* (екранована металевою сіткою).

Важливою характеристикою скрученої пари є категорія. Параметри кабелів різних категорій за стандартом *ISO* 11801 надано у таблиці 2.3.

Таблиця 2.3

Характеристики скрученої пари по категоріях

Позначення категорії	Смуга частот	Максимальна швидкість	Примітка
CAT 5	100 МГц	100 Мбіт/с	Задіяні дві пари
CAT 5*e*	125 МГц	1 Гбіт/с	Задіяні 4 пари
CAT 6	250 МГц	10 Гбіт/с	Тільки до 50 м
CAT 6*a*	500 МГц	10 Гбіт/с	До 100 м
CAT 7	600-700 МГц	10 Гбіт/с	Тільки *S/FTP*

Розглянемо особливості використання волоконно-оптичних кабелів, які набувають все більшого поширення через високу надійність, надвисоку захищеність від прослуховування і можуть забезпечити зростаючі потреби у швидкому передаванні великих обсягів даних в межах земної кулі.

У центральній частині волоконно-оптичних кабелів знаходяться скляні волокна, по яких поширюється світловий промінь. Ці волокна оточені скляною оболонкою, яка має менший показник заломлення, ніж саме волокно, тому світловий промінь віддзеркалюючись від оболонки, повертається у волокно, де

продовжує свій рух далі. В залежності від товщини волокна відрізняють **одномодове** (*Single Mode Fiber, SMF*) зі діаметром від 5 до 10 мкм та **багатомодове** (*Multi Mode Fiber, MMF*), яке має діаметр світловода 50 мкм або 62,5 мкм.

У одномодовому кабелі діаметр світловода є сумірним зі довжиною хвилі світла, яка буває 1,3 або 1,55 мкм. Цим довжинам хвилі відповідає мінімальне ослаблення світлових сигналів. Малий діаметр волокна сприяє руху світла вздовж волокна і зменшує ймовірність віддзеркалювання від оболонки. Тому одномодовий кабель має у десятки і навіть у сотні разів кращі показники якості передавання світлових сигналів, ніж багатомодовий кабель. Сучасні системи на одномодовому волокні працюють зі швидкостями більш ніж 1 Тбіт/с на відстань понад 3000 км.

Перевагою багатомодового кабелю є можливість використовувати випромінювачі на світлових діодах замість лазерних. Це зменшує витрати на обладнання каналів зв'язку, але швидкість і відстань передавання при цьому також зменшуються відповідно до сотень Мбіт/с і декількох кілометрів.

Технології виробництва волоконно-оптичних кабелів постійно вдосконалюються, а їх собівартість зменшується. Такі кабелі виготовляють з різною кількістю волокон (рис. 2.7) від одиниць до десятків і сотень.

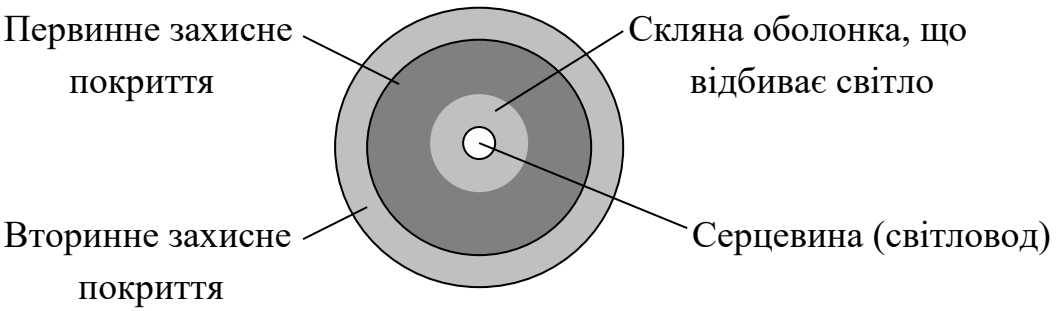


Рис. 2.7. Структура волокна оптичного кабелю

У позначенні кабелю крім кількості волокон прийнято вказувати діаметри серцевини та оболонки (у мікрометрах), наприклад, 9,5/125 або 50/125.

Для переходу від скрученої пари на волоконно-оптичний кабель можна використовувати конвертери – комутатори, що мають тільки два роз'єднувачі різного типу. Широкий асортимент конвертерів дозволяє реалізувати усі можливі варіанти з'єднань різнотипних кабелів.

Роз'єднувачі для волоконно-оптичного кабелю не бажано на довгий час залишати у відкритому вигляді через небезпеку забруднення. Особливо це стосується одномодових кабелів. Існують спеціальні ковпачки для захисту

оптичних контактів під час зберігання. Через те, що промінь світла є дуже тонким, то навіть дрібненька порошинка в роз'єднувачі може погіршити якість зв'язку.

Вартість обладнання волоконно-оптичних мереж може не на багато перевищувати вартість аналогічного обладнання мереж на скрученій парі, але для монтажу оптичних кабелів необхідний досить дорогий інструмент, за допомогою якого можна з'єднувати оптичні волокна методом точного лазерного зварювання. Один зі зразків такого інструменту показано на рис. 2.8.

Рис. 2.8. Апарат для зварювання волокон оптичного кабелю

Не зважаючи на високу вартість цього інструменту, виконання робіт зі монтажу оптичного кабелю можна замовити відносно не дорого.

Внутрішню структуру волоконно-оптичного кабелю для з'єднань в межах закритих приміщень показано рис. 2.9.

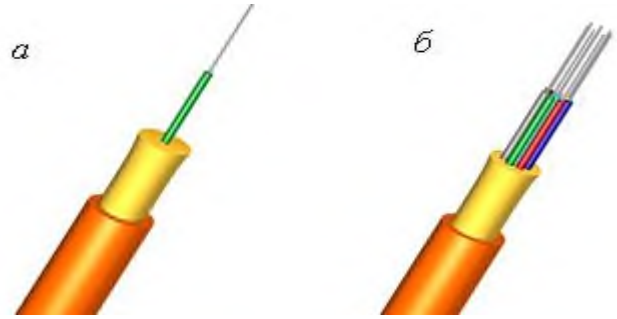


Рис. 2.9. Структура зразків волоконно-оптичних кабелів з одним волокном (*а*) та з вісьма волокнами (*б*), що виконані за конструкцією *Uni Tube* (усі волокна у одній трубі)

В залежності від призначення кабелів їх зовнішнє покриття можуть виготовляти зі різних за міцністю матеріалів, від м’якого пластику до твердої металевої броні. Кабелі з великою кількістю волокон виробляють за конструкцією *Multi Tube* (декілька пластикових труб зі волокнами). У кабелях, що призначені для підвішування на опорах або прокладання у телекомунікаційних каналізаціях чи у землі, обов’язково є силові елементи у вигляді металевих дротів в центрі або по краях, а труби зі волокнами заповнюють гелем. Найтонші гнучкі оптичні кабелі зі одним волокном використовують для виготовлення з’єднувальних шнурів (Рис 2.10).

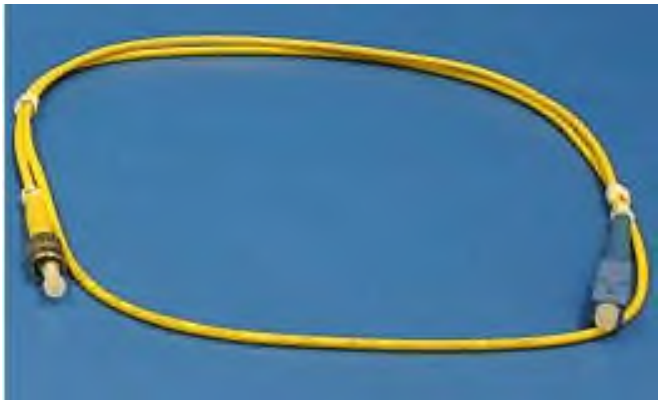


Рис. 2.10. Оптичний з’єднувальний шнур зі роз’єднувачами двох різних типів: *ST* (ліворуч) та *SC* (праворуч)

На кінцях роз’єднувачів є захисні ковпачки (див. рис. 2.10).

Основні типи оптичних роз’єднувачів зображені на рис. 2.11.

а *б* *в* *г*

Рис. 2.11. Зовнішній вигляд оптичних роз’єднувачів різних типів: *а – SMA*; *б – ST*; *в – FC*; *г – SC*.

В перших оптичних технологіях по кожному з волокон сигнали передавали тільки в одному напрямку, але наявність смуг прозорості на частотах 193 ТГц (хвиля 1,55 мкм) та 230 ТГц (1,3 мкм) надала можливість по одномодовому волокну одночасно пересилати дані в обох напрямках (повний дуплекс). Таке обладнання випускають парами (рис. 2.12).

Рис. 2.12. Пара медіа конверторів, які забезпечують повний дуплекс по одномодовому волокну довжиною до 20 км зі швидкістю 100 Мбіт/с (на лівому медіа конверторі бачимо захисні ковпачки від гнізда і шнура)

Щоб не переплутати пари різних комплектів медіа конверторів або парних модулів *SFP*, в кінці позначення кожного з них ставлять латинську літеру *A* або *B*. Ніяких інших зовнішніх розбіжностей парне обладнання не має. Зв'язок встановлюється тільки між пристроями з різними літерами.

Технологія *WDM* (*Wave Division Multiplexing* – мультиплексування зі розподілом по довжині хвилі) дозволяє по одному одномодовому волокну утворювати до 64 незалежних ліній зв'язку, що забезпечує підвищення пропускної здатності раніше прокладених кабелів до 15500 Гбіт/с по кожному з волокон. Існують три різні за ціною варіанти обладнання цієї технології: *CWDM* (*Coarse WDM* – грубе *WDM*), *DWDM* (*Dense WDM* – щільне *WDM*) та *HDWDM* (*High DWDM*), яким відповідають можливості утворення 16, 40 та 64 незалежних ліній по одному волокну.

Існує оптична технологія *POF* (*Plastic Optical Fiber* – пластикове оптичне волокно) з використанням пластикових волокон замість скляних. Пластикові волокна є альтернативою скрученій парі на відстані до десятків метрів, бо такі волокна мають кращі механічні властивості та абсолютно нейтральні до електромагнітних полів. Пластикове оптичне волокно більш гнучке у порівнянні зі скрученою парою, крім того його легко прокладати і підключати за допомогою найпростішого інструменту.

В період з 1999 до 2016 року було прийнято та впроваджено цілий ряд міжнародних стандартів на технології бездротових мереж з використанням

радіо-ефіру. За масштабом (або радіусом дії) бездротові (*Wireless*) технології прийнято розподіляти на наступні три категорії.

- *WPAN – Wireless Personal Area Network* (персональні мережі).
- *WLAN – Wireless Local Area Network* (локальні мережі).
- *WMAN – Wireless Metropolitan Area Network* (регіональні мережі).

Основні характеристики цих технологій надано у таблиці 2.4.

Таблиця 2.4

Характеристики сучасних технологій бездротових мереж

Масштаб	Назва	Максимальна швидкість	Радіус дії
WPAN	*Bluetooth*	24 Мбіт/с	До 15 м
WLAN	*Wi-Fi*	600 Мбіт/с	До 100 м
WMAN	*WiMAX*	1 Гбіт/с	До 10 км

Дані, що наведені у таблиці 2.4, відповідають сучасному стану ринку обладнання бездротових мереж і можуть змінюватись зі появою нових стандартів. Але частоти радіо-ефіру не можуть суттєво змінитись через їх постійний дефіцит. Є помилкою вважати, що, обравши, наприклад, найкращий варіант технології *WiMAX*, можна однією точкою доступу якісно обслуговувати користувачів густо населеного міста в радіусі 10 км. Пропускна здатність цієї точки (1 Гбіт/с) розподіляється між усіма її користувачами. Майже кожен будинок Києва підключено до Інтернету волоконно-оптичним кабелем, що забезпечує швидкість не менше ніж 1 Гбіт/с. Збільшуючи кількість *WiMAX* точок доступу, слід розміщати їх по території так, щоб вони не утворювали завад одна одній, а це в умовах міського насиченого радіо-ефіру не проста задача. Найдоцільніше такі технології використовувати у сільській місцевості, де користувачів не багато і вони знаходяться на значній відстані один від одного. Або можна створювати у містах окремі ділянки зі можливістю бездротового доступу до мережі.

Проблема мобільного доступу, де бездротові технології є незамінними, полягає в тому, що зі збільшенням швидкості пересування втрачається швидкість обміну даними. Розробки стандартів для мобільного доступу мають назву *LTE* (*Long Term Evolution* – довгостроковий розвиток) або 4*G* (четверте покоління). Роботи в цьому напрямку посуваються повільно і навіть

припиняються через безуспішність, але їм на зміну поступово набуває розвитку новий стандарт 5*G* (п'яте покоління).

Зовнішній вигляд адаптера технології 4*G* показано на рис. 2.13.

Рис. 2.13. Адаптер технології 4*G* у порту *USB* пересувного ПК.

Щодо технології 5*G*, яку в Україні було представлено 28 жовтня 2021 року у технопарку *UNIT.City*, то для її широкого впровадження передбачають витратити 1,5-2 роки. Технологія мобільного зв'язку 5*G* приблизно у 10 разів надає більшу швидкість передавання даних, ніж у технології 4*G*. Причина такого зростання пов'язана з використання нового частотного діапазону до 54 ГГц, що дозволило розширити смугу частот передавання і відповідно збільшити пропускну спроможність каналу зв'язку. У технології 4*G* частота не перевищує 6 ГГц. Радіо-інтерфейс для технології 5*G*, який розроблено консорціумом 3*GPP* (*The* 3*rd Generation Partnership Project*), має назву *NR*, що означає *New Radio*. Цей інтерфейс має два нових суттєво віддалених один від одного діапазони частот. Один з них під назвою *FR*1 (*Frequency range* 1) має смугу 100 МГц на частотах орієнтовно від 3 до 6 ГГц. Наприклад, у Південній Кореї для цього діапазону обрали смугу від 3,3 до 4,2 ГГц зі центральною частотою 3,5 ГГц. Другий діапазон них під назвою *FR*2 (*Frequency range* 2) займає смугу від 50 до 400 МГц на частотах від 24 до 54 ГГц. Усі ці частоти раніше не використовували для телефонії та Інтернету, але на деяких з них ще працюють у авіації та супутникових системах, що потребує часу для міжвідомчого узгодження. Крім того сигнали на частотах діапазону *FR*2 добре розповсюджуються лише у вільному просторі. Навіть опади у вигляді дощу або снігу суттєво ослаблюють потужність сигналу. Хоч антени у новій технології мають менші розміри, а технічне обладнання має меншу вартість, але для забезпечення високої якості зв'язку потрібно набагато більше точок доступу. Краще за все, щоб такі точки

були розміщені у кожній кімнаті. Але невпинно зростаючі потреби людства у якості мобільного Інтернету свідчать про невідворотність широкого впровадження технології 5*G*. При цьому переваги та особливості її впровадження полягають у наступному.

- Можливість одночасного обслуговування 10 тис. абонентів на площі близько 1 га. Це надає можливість якісного доступу до мережі під час зібрання великої кількості людей, а також у системах Інтернету речей (*IoT*) за умов скупчення багатьох об'єктів управління.
- Адаптивне керування модуляцією та кодуванням (*MCS – modulation and coding scheme*) надає можливість зменшувати швидкість передачі для недопущення зростання кількості помилок або втрати зв'язку.
- Передбачена можливість бездротового передавання енергії для живлення пристроїв. Це може бути корисним у системах Інтернету речей там де виникають ускладнення з прокладанням кабелю.
- Мобільні пристрої, що підтримують технологію 4*G* не підійдуть для технології 5*G*.

2.2.3. Технології канального рівня

Призначення кожної з численних технологій канального рівня полягає в утворенні каналу передачі даних між певною кількістю вузлів, які поєднуються в мережу за цією технологією. Усі технології канального рівня побудовані з використанням одного або декількох типів обладнання фізичного рівня. Різноманіття технологій канального рівня пов’язано зі різною віддаленістю та кількістю вузлів, які необхідно об’єднати в мережу, а також з різними вимогами до швидкості, надійності та достовірності передавання даних.

Найбільш універсальною в даний час можна вважати технологію *Ethernet*, яка фактично являє собою сім’ю технологій, котрі поєднує єдиний принцип доступу до фізичного середовища та спільні правила формування пакетів. Є варіанти технології *Ethernet*, які дозволяють з’єднувати між собою пари вузлів на будь-якій відстані, що потрібно для побудови мереж глобального масштабу, а також є варіанти, що дозволяють об’єднати до 1024 вузлів в мережу локального масштабу [7-9]. Значення швидкості передавання даних для різних варіантів технологій сім’ї *Ethernet* надано у таблиці 2.5.

Таблиця 2.5

Варіанти технологій сім'ї *Ethernet* з різною швидкістю передавання

Назва технології	Максимальна швидкість	Рік випуску стандартів
Ethernet	10 Мбіт/с	1981 - 1991
Fast Ethernet	100 Мбіт/с	1995
Gigabit Ethernet (*GbE*)	1 Гбіт/с	1998-2001
10 *Gigabit Ethernet* (10*GbE*)	10 Гбіт/с	2002-2007
40 *Gigabit Ethernet* (40*GbE*)	40 Гбіт/с	2010
100 *Gigabit Ethernet* (100*GbE*)	100 Гбіт/с	2010
400 *Gigabit Ethernet* (400*GbE*)	400 Гбіт/с	2016

Знайомлячись з літературними джерелами, а також з інформацією, яка надається у мережі Інтернет, слід перевіряти дати видання і з обережністю ставитись до думок, що можуть втратити актуальність. Технічний прогрес іноді приносить великі несподіванки. Це можна спостерігати на прикладі технології *ATM* (*Asynchronous Transfer Mode*), про яку на ресурсі http://uk.wikipedia.org/wiki сказано, що прибутки від продажу обладнання цієї технології у 1997 році склали 2,4 млрд. доларів, у 1998 році – 3,5 млрд. доларів, а на 2001 рік прогнозували зростання до 9,5 млрд. Вважали, що *ATM* найперспективніша технологія для майбутнього. Обставини змінились на початку 21-го сторіччя через початок зникнення дефіциту пропускної спроможності фізичних ліній зв'язку. До того часу зростання пропускної спроможності стримувалось через застарілість технологій побудови ліній зв'язку. Завдяки досягненням розробників волоконно-оптичних кабелів, стало можливим створення високопродуктивних ліній зв'язку. Тому технології, які призначені для роботи в умовах дефіциту пропускної спроможності, наприклад, *ATM*, втратили актуальність.

В загальному вигляді задача технології канального рівня полягає в тому, щоб забезпечити передачу порцій даних від вузла відправника до вузла одержувача в межах однієї мережі. Кількість вузлів в цій мережі найчастіше не перевищує десятків, а у режимах, які мають назву точка-точка, поєднуються тільки два вузли. Порції даних (пакети) на канальному рівні усіх сучасних комп'ютерних мереж прийнято називати *frame* (кадр). Розмір цих порцій завжди обмежений. Для технологій сім'ї *Ethernet* це обмеження становить 1500 байт.

Головною вимогою до технологій канального рівня є забезпечення потрібної швидкості доставки даних, можливо ще й з не перевищенням допустимої затримки кожного пакету. Крім того існує норма на можливі втрати пакетів, яка в Україні дорівнює 0,1%. Оскільки під час передавання можуть виникати помилки, то в усіх технологіях канального рівня передбачені перевірки за допомогою контрольних сум. У разі, коли така перевірка дає негативний результат, весь пакет знищують. Продуктивність канального рівня в першу чергу залежить від якості фізичного середовища передавання сигналів. Зрозуміло, коли з'явились якісні волоконно-оптичні кабелі, то якраз найпростіша і найдешевша технологія, якою є *Ethernet* виявилась найдоцільнішою. Так буде скоріш за все до чергового виникнення дефіциту пропускної спроможності каналів зв'язку.

Розглянемо головні особливості технологій сім'ї *Ethernet*.

В основу технології *Ethernet* покладено принцип невпорядкованості в часі моментів початку передавання даних для всіх вузлів мережі в умовах єдиного спільного середовища передавання сигналів. Цей принцип був скопійований зі мережі *Aloha*, яку ще на початку 70-х років було створено у Гавайському університеті США. Спільним середовищем передавання у мережі *Aloha* був радіо-ефір. Такий метод, коли кожен вузол мережі може починати передачу коли завгодно, має назву *CSMA/CD* (*Carrier Sense Multiple Access with Collision Detection* – груповий доступ з контролем носія та виявленням колізій). Зрозуміло, що невпорядкованість в часі призводить до можливих конфліктів, а саме тоді, коли двоє або більше вузлів майже одночасно починають передавати сигнали у спільне середовище. Ці ситуації було названо колізіями і подальший шлях розвитку таких мереж був пов'язаний зі розробкою методів подолання колізій.

Ймовірність виникнення колізій в значній мірі залежить від завантаженості середовища передавання, а точніше від співвідношення між кількістю даних, які необхідно передати і пропускною спроможністю каналу зв'язку. З'ясовано, що для успішного подолання колізій необхідно, щоб середовище передавання було завантажене не більше ніж приблизно на третину від своїх можливостей. Зв'язківці дуже негативно сприйняли такі показники, бо завантаженість середовища передавання завжди була одною з основних ознак ефективності систем зв'язку. У той же час були створені системи позбавлені від колізій (*Token ring, ARCNET*). В таких умовах розпочиналась діяльність по створенню технології *Ethernet*.

Ідея створення технології *Ethernet* належить Роберту Меткалфу, який заснував компанію 3*Com* з метою розвитку і впровадженню своєї ідеї. Йому вдалося досягти домовленості зі потужними компаніями *DEC, Intel* і *Xerox* про

спільну роботу над стандартом *Ethernet* (*DIX*). Стандарт вийшов в світ 30 вересня 1980 року. Це був важливий крок до майбутніх перемог над популярними у той час технологіями *Token ring* і *ARCNET*. Починаючи з 1980 року усі поновлення у сім'ї технологій *Ethernet* погоджуються зі комітетом 802 *IEEE*.

Усі вузли мережі *Ethernet* підключені до спільного середовища передавання сигналів і мають право у будь-який момент починати передачу при умові що не прослуховується чужий сигнал. Кожний вузол весь час прослуховує сигнал. При відсутності колізії сигнал, що передається, повинен співпадати зі сигналом, що прослуховується. У разі виявлення розбіжності припиняється передавання пакета, і замість нього передається спеціальний сигнал (*jam*-послідовність), який відрізняється від інформаційних пакетів і повідомляє усі вузли мережі про перехід у режим затримання початку передавання на випадковий проміжок часу. Після цього усі вузли, що мають розпочати передавання або вже розпочали передавання, повинні витримати паузу тривалістю у m інтервалів довжиною 512 проміжків між бітами, де m – ціле число, що вибирають з рівною ймовірністю у діапазоні $[0, 2^N]$, де N – номер спроби передавання даного пакета, але не більше за 10. Після десятої спроби довжину діапазону не збільшують, а після 16 спроб невдалого подолання колізії припиняють передачу цього пакета.

Передавання кожного кадру розпочинається з 64-бітової (8 байт) преамбули, що являє собою таку послідовність:
10101010101010101010101010101010101010101010101010101010101011

Зверніть увагу на дві одиниці у кінці преамбули. Ця послідовність дає змогу налаштувати систему синхронізації приймача і виявити момент початку заголовку кадру. Наявність преамбули нагадує частоту носія, яку вмикають перед початком передавання сигналів в радіо-ефір. Хоч назва технології *Ethernet* складається зі слів *Ether* (ефір) та *net* (мережа), але у перших стандартах використання радіо-ефіру не передбачалось, що видно з таблиці 2.6.

Таблиця 2.6

Перші стандарти технології *Ethernet* для різних середовищ передавання зі швидкістю 10 Мбіт/с

Назва стандарту	Рік вип.	Середовище передавання	Довжина пасивного з'єднання
IEEE 802.3 (10*BASE*5)	1983	Коаксіальний кабель RG8 (товстий) ∅ 9,5 мм	<500 м
IEEE 802.3*a* (10*BASE*2)	1985	Коаксіальний кабель RG58 (тонкий) ∅ 5 мм	<185 м
IEEE 802.3*i* (10*BASE-T*)	1990	Скручена пара *UTP cat*.3	<100 м
IEEE 802.3*j*(10*BASE-F*)	1993	Волоконно-оптичний	<2000 м

Деякі стеки протоколів мають вимоги до канального рівня вищі у порівнянні зі стеком *TCP*/*IP*, наприклад, втрата пакетів не дозволяється. Для цього було розроблено більш складну структуру кадрів, яка зображена на рис. 2.14. (Найпростішу структуру кадрів було представлено на рис. 1.10)

802.3/*LLC*

DA	*SA*	*L*	*DSAP*	*SSAP*	*C*	*Data*	*FCS*
6	6	2	1	1	1(2)	46-1497(1496)	4

Ethernet SNAP

DA	*SA*	*L*	*DSAP*	*SSAP*	*C*	*OUI*	*T*	*Data*	*FCS*
6	6	2	1	1	1	3	2	46-1492	4

Заголовок *MAC* | Заголовок *LLC* | Заголовок *SNAP*

Рис. 2.14. Формати кадрів *Ethernet* для різних стеків протоколів

Під позначкою кожного поля кадру наведено довжину у байтах.

DA – *Destination Address* – адреса одержувача.

SA – *Source Address* – адреса відправника.

L – *Length* – довжина поля даних у байтах.

T – *Type* – має те ж значення, як у кадрі *Ethernet DIX* (див. рис.1.10) .

Data – поле даних, які пересилаються у цьому кадрі.

FCS – *Frame Check Sequence* – контрольна сума.

Додатковий заголовок *LLC*, який ми бачимо у цих кадрах, дозволяє керувати логікою роботи протоколу *Ethernet*. Цей заголовок вміщує такі поля:

DSAP – Destination Service Access Point – код точки доступу до служби одержувача, що визначає ту чи іншу програму обробки пакета;

SSAP – Source Service Access Point – код точки доступу до служби відправника;

C – Control – управління, що має три варіанти структури, які зображено на рис. 2.15.

Для ненумерованих кадрів (*Unnumbered*)

1	1	*M*	*P/F*	*M*
1	1	2	1	3

Для керуючих кадрів (*Supervisory*)

1	0	*S*	–	*N* (*R*)
1	1	2	5	7

Для інформаційних кадрів (*Information*)

0	*N* (*S*)	*P/F*	*N* (*R*)
1	7	1	7

Рис. 2.15. Варіанти структури поля управління

Під позначенням (або змістом) кожної ділянки поля надано кількість бітів, а інформація у них може бути занесена така:

M – тип команди;

S – службова інформація;

P/F – ознака того, що потрібна відповідь на команду;

N(*S*) – номер кадру, що відправлений;

N(*R*) – номер кадру, що очікується.

Якщо повідомлення займає більше ніж 128 кадрів, нумерація продовжується за циклом.

Заголовок *SNAP* (*SubNetwork Access Protocol* – протокол доступу до підмереж) складається з двох полів:

OUI – Organizationally Unique Identifier – код організації (установи);

Управління логікою передавання *LLC* полягає у можливості вибору одного з трьох наступних режимів:

*LLC*1 – без встановлення з’єднання та без підтвердження;

*LLC*2 – із встановленням з’єднання та з підтвердженням;

*LLC*3 – без встановлення з’єднання, але з підтвердженням.

Якщо підтвердження не отримано, кадр відправляють повторно.

Різні режими управління логікою передавання потрібні для того, щоб забезпечити можливість роботи з усіма стандартами стеків протоколів.

2.2.4. Особливості технологій сім'ї *Ethernet*

Обраний Робертом Меткалфом метод *CSMA/CD* накладає обмеження на масштаб мережі через можливість появи не виявлених колізій. Умовою працездатності такої мережі є чітка робота механізму виявлення колізій, який проілюстровано на рис. 2.16, де помічені наступні моменти часу:

t_1 – початок передавання пакету першим вузлом;

t_2 – початок передавання пакету другим вузлом;

t_3 – момент виявлення колізії на другому вузлі і початок передавання *jam*-послідовності (сигнал з 32 бітів, який сповіщає про колізію);

t_4 – момент прийняття *jam*-послідовності першим вузлом.

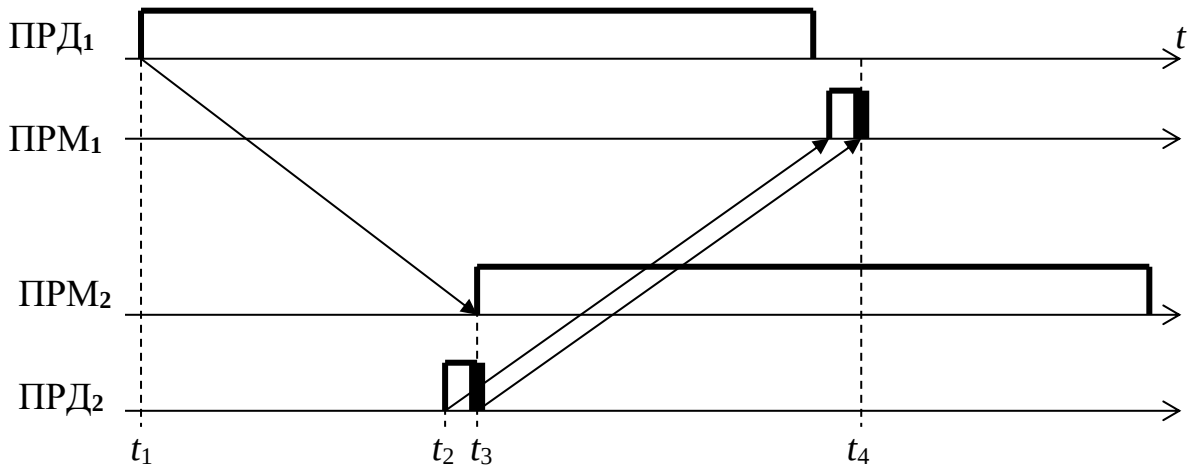


Рис. 2.16. Процеси на передавачах (ПРД) і приймачах (ПРМ) двох найбільш віддалених вузлів під час виявлення колізії

З цього рисунку бачимо, що в інтервалі тривалістю $t_4 - t_1$ можна переслати тільки один пакет. Цей інтервал має назву *PDV* (*Path Delay Value* – найбільша затримка обертання). Іншими словами це подвійна затримка сигналу в каналі зв'язку між найбільш віддаленими вузлами мережі. Якщо після початку передавання сигналу вузол не отримав *jam*-послідовність протягом *PDV*, то це свідчить про відсутність колізії. Чім ближче вузли від передавача, тим раніше прибуває повідомлення про колізію, але необхідно продовжувати паузу через можливість дочекатись *jam*-послідовність від найвіддаленішого вузла. Це і є

інтервал *PDV*. Через його наявність маємо обмеження на діаметр і масштаб мережі.

Після передавання кожного кадру є пауза для забезпечення можливості обробки даних приймачем і підготовки до прийняття наступного кадру. Тривалість цієї паузи дорівнює 96 бітовим інтервалам. Оскільки в усіх варіантах технології *Ethernet* нормують мінімальну довжину кадру, то для розрахунку діаметру мережі використовують формулу $PDV < T_{min} + T_p$, де T_{min} – мінімальна тривалість кадру, T_p – тривалість технологічної паузи. За цим розрахунком діаметр мережі *Fast Ethernet*, де мінімальна тривалість кадру разом зі преамбулою складає 592 бітових інтервалів, не може перевищувати 1000 м при умові, що сигнал в кабелі розповсюджується зі швидкістю світла, а затримок на обробку сигналів не існує. Рекомендований за стандартом діаметр такої мережі зі врахуванням затримок становить 400 м, що відповідає виключно локальним мережам. Важливий крок у розвитку технології *Ethernet* полягє у розподілі середовища передавання на окремі домени колізій з поступовою відмовою від коаксіального кабелю. При цьому на заміну концентраторам, що підсилювали та розгалужували сигнали, прийшли комутатори, які у значній мірі позбавили мережу від колізій і дозволили суттєво збільшити її діаметр. Шлях до глобальних мереж було відкрито зі появою режиму повного дуплексу, де зв’язок відбувається між двома вузлами по двох незалежних трактах передавання сигналів в кожному напрямку. Зрозуміло, що колізії тут неможливі, бо в кожному з трактів є тільки один передавач, при цьому відстань передавання стає практично не обмеженою.

Розглянемо процедури перетворення послідовності бітів у сигнали фізичного рівня мереж *Ethernet*. Комутатори технології *Fast Ethernet* можуть підтримувати 5 режимів роботи:

- 10*Base-T* – по двох скручених парах категорії 3;
- 10*Base-T full-duplex* – по двох скручених парах категорії 3;
- 100*Base-TX* – по двох скручених парах категорії 5;
- 100*Base-T*4 – по чотирьох скручених парах категорії 3;
- 100*Base-TX full-duplex* – по двох скручених парах категорії 5.

Вибір того чи іншого режиму здійснюється шляхом спеціальних переговорів (*Auto-negotiation*) між пристроями канального рівня. Сценарій цих переговорів побудовано таким чином, щоб нові пристрої, які мають більшу кількість режимів, мали б можливість налагодити зв’язок зі старим обладнанням мереж *Ethernet*.

Для передавання послідовності бітів використовують різні варіанти імпульсного кодування. Кілька таких варіантів зображено на рис.2.17.

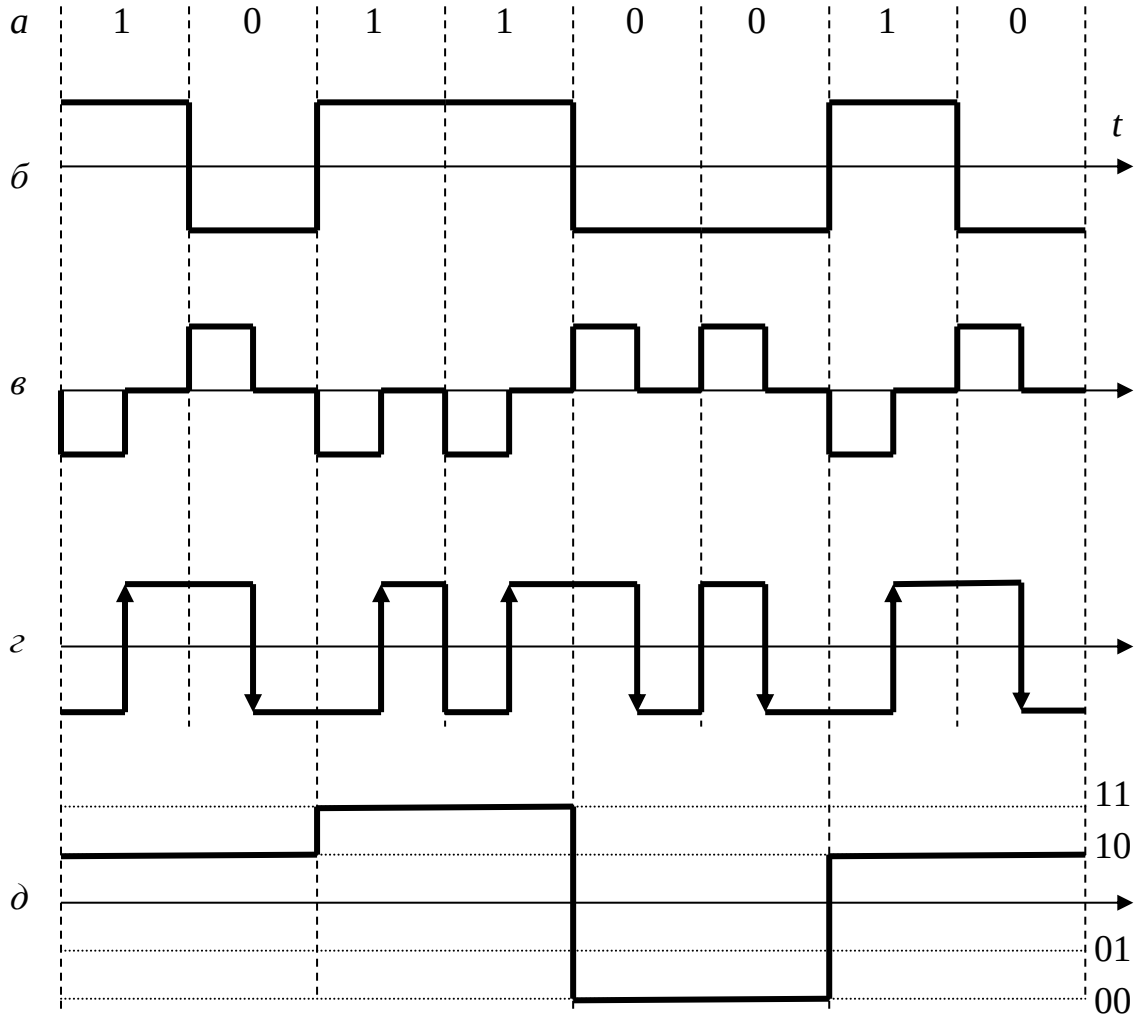


Рис.2.17. Методи кодування у каналах зв’язку:
а – вихідна послідовність бітів; *б* – код *NRZ* (*Non Return to Zero*);
в – код *RZ* (*Return to Zero*); *г* – Манчестерський код; *д* – код *PAM*-5

Розглянемо особливості кожного з цих методів кодування.

У найпростішому методі, що зветься *NRZ* (без повернення до нуля), використовують два рівні потенціалу (електричної напруги або інтенсивності світлового променя). Верхній рівень означає 1, а нижній – 0. Недоліком цього методу є труднощі у синхронізації, що виникають під час передавання довгих послідовностей нулів або одиниць.

Тривалість передавання кожного біта на швидкості 10 Мбіт/с становить 0,1 мкс, а у кадрі може налічуватись до 12 тисяч бітів. Протягом цього часу треба визначати моменти для оцінювання кожного сигналу. Щоб забезпечити необхідну точність визначення таких моментів треба коригувати частоту синхронізації. Для цього використовують моменти зміни потенціалу. Дані про відхилення фактичного моменту зміни потенціалу від того, що визначено

таймером синхронізації, є інформацією для коригування. Відсутність зміни потенціалу під час передавання довгої послідовності нулів або одиниць не дозволяє коригувати частоту, що може призвести до втрати точності синхронізації. У цьому й полягає недолік методу *NRZ*.

У коді *RZ* першу половину бітового інтервалу займає імпульс, що несе інформацію (негативний імпульс означає 1, а позитивний – 0), а другу половину – нульовий потенціал. Перевагою такого методу кодування є простота синхронізації, але при цьому спектр сигналу буде удвічі ширший, ніж для коду *NRZ*, що пояснюється у додатку 5.

Манчестерський код, що довгий час був єдиним кодом фізичного рівня для мереж *Ethernet*, відрізняється найпростішою процедурою синхронізації, бо зміна потенціалу обов'язково відбувається на кожному бітовому інтервалі. Напрямок цієї зміни (у середині бітового інтервалу), що зображено стрілками на рис. 2.21 *г*, обрано за інформаційну ознаку. Зміна з нижнього на верхній рівень означає 1, а з верхнього на нижній – 0. У цьому коді використовується два рівні потенціалу, що забезпечує більшу завадостійкість при однаковій максимальній потужності сигналу, ніж у коді *RZ*. Сигнали при цьому є протилежними, що відповідає максимальній завадостійкості. Фактично тут використовується фазова модуляція з двома варіантами фази сигналу.

З метою підвищення швидкості передавання даних та збільшення продуктивності використання смуги частот каналу у технологіях *Fast Ethernet* та *Gigabit Ethernet* використовують код *PAM*-5 (5-*level Pulse Amplitude Modulation* – п'ятирівнева амплітудно-імпульсна модуляція). При цьому нове обладнання підтримує роботу у манчестерському коді для налагодження зв'язку зі застарілим обладнанням, а також для узгодження в автоматичному режимі роботи за технологією *NWay Auto-Negotiation*, яку ми розглянемо нижче.

У технології *Fast Ethernet* для передавання інформації зі швидкістю 100 Мбіт/с використовують код 4B/5B, який полягає в тому, що кожні 4 біти вихідного коду замінюють на комбінацію з 5 бітів, користуючись таблицями відповідності, що показані на рис. 2.18.

4 біти	5 бітів
0000	11110
0001	01001
0010	10100
0011	10101

4 біти	5 бітів
0100	01010
0101	01011
0110	01110
0111	01111

4 біти	5 бітів
1000	10010
1001	10011
1010	10110
1011	10111

4 біти	5 бітів
1100	11010
1101	11011
1110	11100
1111	11101

Рис. 2.18. Відповідність 4 і 5 бітових послідовностей для коду 4B/5B

Після такого перекодування виключається можливість появи довгих послідовностей нулів або одиниць, що дозволяє використовувати код *NRZ*. При цьому для забезпечення швидкості передавання інформації 100 Мбіт/с необхідно передавати біти зі швидкістю 125 Мбіт/с, що не призводить до суттєвого розширення спектра сигналу і може бути забезпечено на кабелі типу скрученої пари категорії 5.

У технології *Gigabit Ethernet* для передавання інформації із швидкістю 1000 Мбіт/с використовують код *PAM*-5, у якому кожним двом бітам відповідає один з п'яти рівнів потенціалу. При цьому тривалість імпульсів вибрано 8 нс (такою ж як в технології *Fast Ethernet*). Це дозволяє досягти швидкість у 250 Мбіт/с на кожній парі категорії 5. На чотирьох скручених парах, що використовують одночасно, максимальна швидкість дорівнює 1000 Мбіт/с.

У мережах *Ethernet* можуть взаємодіяти декілька різношвидкісних технологій. Розглянемо як це відбувається за допомогою технології *NWay Auto-Negotiation.*

Партнерами операції узгодження режимів роботи є *Ethernet*-порти різного обладнання. Кожен партнер повідомляє іншому про технологію, яку він підтримує, надсилаючи імпульсні послідовності у манчестерському коді з періодом 16,8 мс. Найстаріше обладнання, що підтримує тільки один режим 10Base-T, надсилає лише один імпульс *NLP* (*Normal Link Pulse*), що свідчить тільки про його працездатність. Обладнання *Fast Ethernet* та *Gigabit Ethernet* надсилають інформаційне слово *LCW* (*Link Code Word*), структура якого наведена у таблиці 2.7.

Таблиця 2.7

Призначення полів інформаційного слова *LCW*

Найменування поля	Кількість бітів	Призначення поля
Селектор	5	Код базової технології
Технологічні можливості	8	Код режиму роботи
RF (*Remote Fault*)	1	Прапорець, що повідомляє про помилку
Ack	1	Прапорець підтвердження
NP (*Next Page*)	1	Прапорець наявності додаткової інформації (продовження)

Технологія узгодження передбачає розширення у разі появи нових базових технологій та режимів роботи.

Процедура узгодження полягає у виборі найбільш пріоритетного з можливих режимів роботи. Пріоритети обрано таким чином, щоб режимам з більшою швидкістю передавання відповідали вищі пріоритети.

Найбільші пріоритети надані дуплексним режимам. Це такі режими, що разом з появою комутаторів, фактично призвели до суттєвих змін у технології *Ethernet*. Розглянемо детальніше ці зміни.

Основою створення технології *Ethernet* було спільне середовище передавання з неминучістю колізій. На початку цим середовищем був радіоканал (ефір). Далі було впроваджено коаксіальний кабель, що виконував функції ефіру, залишаючись спільним середовищем. Поява концентраторів полегшила обслуговування мережі, але не позбавила від спільного середовища з колізіями. Все це примушувало враховувати суворі обмеження на максимальну відстань між вузлами.

Дуплексний Ethernet використовує два окремі фізичні середовища передавання між парою вузлів. При цьому одночасне передавання з обох вузлів не призведе до колізії, бо кожному з напрямків передавання надається окреме середовище. Ніяких обмежень на відстань, що пов’язані з можливістю колізій, у дуплексному режимі не існує.

Комутатор, хоч і нагадує за зовнішніми ознаками концентратор, але він виконує розподіл середовища передавання на окремі сегменти. Кожен порт комутатора має окремий процесор з блоком пам’яті на один або декілька кадрів. Сигнали від вузлів мережі потрапляють не у спільне середовище, а у пам’ять свого сегмента. Зрозуміло, що колізії при цьому неможливі. Центральний процесор комутатора, що забезпечує обмін даними між сегментами, має високу швидкодію. Він запам’ятовує фізичні адреси кожного вузла та пересилає пакети між сегментами з врахуванням адрес одержувачів. Якщо адреса невідома, пакет надсилається на всі вузли. Поступово таблиця адрес у комутаторі доповнюється, бо у кожному пакеті, що потрапляє на який завгодно порт комутатора, є фізична адреса відправника. Ця адреса запам’ятовується разом з номером порту, з якого надійшов цей пакет. Різні порти комутатора можуть приймати та передавати пакети з різною швидкістю, бо швидкість, з якою було прийнято пакет на одному з портів, не залежить від швидкості, з якою цей пакет буде передано з іншого порту.

В и с н о в к и

1. Канал зв’язку – це сукупність середовища для передачі сигналів, передавача і приймача сигналів. Під сигналом розуміють фізичний процес, що використовують для передачі інформації. Інформацію в комп’ютерних мережах прийнято називати даними, тому канали в цих мережах називають каналами передачі даних.
2. Вся комп’ютерна інформація зберігається і обробляється у вигляді послідовностей двійкових одиниць, які позначають цифрами 0 та 1. Таку форму представлення інформації називають дискретною або цифровою. Сигнали, якими передають дискретну інформацію, також називають дискретними. Кількість варіантів цих сигналів обмежена.
3. Більшість процесів у нашому житті є безперервними, а для обробки в комп’ютері їх перетворюють у дискретні. Після обробки можна надати результату безперервний вигляд. Ці перетворення називають аналогово-дискретними (або дискретизацією) та дискретно-аналоговими. Важливою задачею дискретизації є вибір частоти вимірів безперервного процесу. Цю задачу було розв’язано відомим вченим В.О.Котельниковим.
4. У комп’ютерних мережах сигнали являють собою імпульсні процеси (імпульси). Під імпульсними розуміють процеси, тривалість яких сумірна з тривалістю нестаціонарних процесів у середовищі передавання. Для збільшення швидкості передачі скорочують імпульси, але при цьому їх стає важко розпізнавати. Доведено, що максимальна частота передавання імпульсних сигналів не може перевищувати $2F$, де F – смуга частот передавання.
5. На сигнали впливають детерміновані та випадкові фактори. До детермінованих відносять перетворення форми через відхилення амплітудно-частотних та фазово-частотних характеристик від ідеальних. Випадкові фактори (завади) являють собою процеси, параметри яких описують за допомогою методів теорії ймовірностей.
6. У багатьох каналах зв’язку смуга частот не містить низькочастотних складових. В першу чергу це стосується радіозв’язку. При цьому неможливо пересилати імпульси, спектр яких починається з частоти, що наближається до нуля. Виникає необхідність у спектральному перетворенні сигналів (модуляції), яке можна представити у вигляді процедури множення імпульсу, на синусоїду, що має частоту у потрібному діапазоні.

7. Розробники стеку *TCP/IP* не розглядали окремо фізичний та канальний рівні, бо задача, яку вони вирішували, полягала у створенні протоколів вищих рівнів, а до нижніх рівнів їм необхідно було тільки розробити інтерфейси. Модель *ISO/OSI* запропонована виходячи з можливості незалежного розв'язання задач на кожному з рівнів. Розподіл на фізичний та канальний рівні у моделі *ISO/OSI* є чітко визначеним.
8. Фізичний рівень охоплює характеристики пристроїв, які необхідні для приєднання до середовища, і характеристики процесів, які використовують для передачі інформації. На цьому рівні визначають форму, розміри та інші параметри роз'єднувачів та антен, а також потужність, частоту та інші характеристики сигналів. Задачі фізичного рівня полягають в забезпеченні надійного інтерфейсу з середовищем, створення засобів приєднання до середовища і вибір сигналів, які б забезпечили потрібну швидкість передачі даних.
9. На канальному рівні обумовлюються правила формування кадрів (пакетів канального рівня). Це формати заголовків та кінцівок кадрів і обмеження, що накладаються на кількість та порядок розміщення даних в кадрі. Також визначаються алгоритми передавання кадрів.
10. Найбільш популярний для побудови ЛКМ кабель типу *UTP* (неекранована скручена пара) є добре захищеним від завад. Захист забезпечується тим, що проводи кожної пари скручують між собою, а сигнали формують симетрично таким чином, щоб полярність імпульсів в проводах пари була протилежна. Вплив завад при цьому суттєво зменшується. Такі сигнали називають диференціальними.
11. Виготовляють одножильний і багатожильний кабель. Одножильний призначений для прокладання у коробах або закріплення на стінах. На його кінцях встановлюють розетки. Багатожильний призначений для виготовлення з'єднувальних шнурів з вилками на кінцях. Екрановані і неекрановані кабелі мають різний хвильовий опір (150 Ом і 100 Ом – відповідно). Незважаючи на те, що екрановані і неекрановані гнізда однакові за розміром, вони призначені тільки для конкретного типу кабелю.
12. Волоконно-оптичні кабелі бувають одномодові (діаметр жили від 5 до 10 мкм) та багатомодові (діаметр світловода 50 мкм або 62,5 мкм). Малий діаметр волокна сприяє руху світла вздовж волокна і зменшує ймовірність віддзеркалювання від оболонки. Тому одномодовий кабель має у десятки і

навіть у сотні разів кращі показники якості передавання світлових сигналів, ніж багатомодовий кабель.

13. Технології побудови бездротових мереж з використанням радіо-ефіру прийнято розподіляти на три категорії: персональні мережі (радіус до 15 м), локальні мережі (радіус до 100 м) та регіональні мережі (радіус до 10 км).
14. Технологія *Ethernet* є найбільш універсальною для побудови каналів передачі даних для мереж різного масштабу. Вона фактично являє собою сім'ю технологій, котрі поєднує єдиний принцип доступу до фізичного середовища та спільні правила формування кадрів. Є варіанти технології *Ethernet*, які дозволяють з'єднувати між собою пари вузлів на будь-якій відстані, а також є варіанти, що дозволяють об'єднати до 1024 вузлів в локальну мережу. Швидкості передавання даних для різних варіантів *Ethernet* можуть бути 10, 100, 1000 Мбіт/с та 10, 40, 100, 400 Гбіт/с.
15. Для Інтернету все ширше використовують канали мобільного зв'язку. З кожною новою генерацією технологій такого зв'язку якість доступу до мережі Інтернет суттєво покращується. Особливо це відчувається з появою технологій 4*G* та 5*G*.

Запитання та завдання для самоперевірки

1. Надайте визначення поняттям сигнал, канал зв'язку та канал передачі даних.
2. Для чого необхідні аналогово-дискретні та дискретно-аналогові перетворення?
3. Яка відповідність між тривалістю імпульсних сигналів, їх спектром та смугою частот передавання?
4. Які задачі канального та фізичного рівнів моделі *ISO/OSI*?
5. Чим забезпечують захист від завад у з'єднаннях кабелем типу *UTP*?
6. Яке призначення одножильних та багатожильних кабелів?
7. В чому особливості використання екранованої та неекранованої скрученої пари?
8. Які різновиди та особливості волоконно-оптичних кабелів?
9. В чому полягають особливості використання радіо-ефіру для побудови комп'ютерних мереж?
10. Опишіть особливості та переваги технології *Ethernet*.
11. Які діапазони частот і з якою метою використовують у технології 5*G*.

Р О З Д І Л 3

ОБ'ЄДНАННЯ КОМП'ЮТЕРНИХ МЕРЕЖ

3.1. Концепції складних комп'ютерних мереж

Усі технології канального рівня мають обмеження у кількості вузлів, а також можуть мати обмеження у відстані між найбільш віддаленими вузлами (діаметр мережі). Наприклад, для мереж, що побудовані за технологією *Ethernet*, максимальна кількість вузлів не повинна перевищувати 1024. Через ці обмеження виникла потреба у об'єднанні мереж, щоб надати можливість обміну даними між вузлами різних мереж, що побудовані за будь-якими технологіями канального рівня.

Об'єднання мереж дозволяє утворити необмежену за розмірами та кількістю вузлів мережу. При цьому вузли повинні ідентифікуватись за допомогою спеціальних адрес об'єднаної мережі. Ці адреси надаються за протоколом *IP* (*Internet Protocol* – протокол зв'язку між мережами) і їх прийнято називати *IP* (ай-пі) адресами. Загальна кількість адрес для четвертої версії *IP* протоколу *IPv*4 перевищує 4 мільярди (2^{32}), що до червня 2012 року вистачало користувачам Інтернету. Але з 6 червня 2012 року майже всі магістральні мережі переведено на шосту версію *IPv*6, у якій кількість адрес практично необмежена. Їх може бути 2^{128}, що перевищує кількість грамів речовини земної кулі.

Крім надання унікальних адрес вузлам складної мережі необхідно ще відшукувати шлях (маршрут) передачі даних від вузла відправника до вузла одержувача крізь канали різних мереж. Цю задачу вирішують вузли-маршрутизатори, які одночасно є вузлами двох або більшої кількості окремих мереж.

Задача маршрутизації полягає в пошуку найкращого маршруту доставки пакетів, але при цьому не гарантується успішність доставки, бо за різних причин пакети можуть бути втрачені. Для забезпечення якісного зв'язку між вузлами необхідно мати наскрізне і прозоре сполучення, що означає збереження послідовності бітів під час передавання і відсутність будь яких заборонених комбінацій у цій послідовності.

Задачу утворення наскрізного прозорого сполучення розв'язує протокол транспортного рівня, який має назву *TCP* (*Transmission Control Protocol* – протокол управління передачею).

3.2. Протокол *IP*

Єдиним серед усіх протоколів мережі Інтернет, без використання якого не може обійтись жоден сеанс передавання даних є протокол *IP*, який вважають головним протоколом Інтернету [10-12].

Структуру заголовку пакетів *IPv*6 показано у таблиці 3.1.

Таблиця 3.1

Структура заголовку пакета *IPv*6

Найменування параметру	Кількість біт	Значення параметру
Номер версії (*Version*)	4	0110 (версія 6)
Клас трафіку (*Traffic Class*)	8	Перші 3 біти визначають клас трафіку, а решта визначає пріоритет
Мітка потоку (*Flow Label*)	20	Псевдовипадкове число, яке не змінюють в межах потоку даних
Довжина даних (*Payload Length*)	16	Кількість байт даних у пакеті (без заголовку)
Наступний заголовок (*Next Header*)	8	Номер протоколу, заголовок якого є наступним (6 – *TCP*, 17 – *UDP*)
Ліміт пересилань (*Hop Limit*)	8	Кількість маршрутизаторів, що може пройти пакет до моменту знищення
IP-адреса відправника пакета (*Source*)	128	
IP-адреса одержувача пакета (*Destination*)	128	

Параметр мітка потоку призначений для прискорення маршрутизації. Він зберігається в пам’яті маршрутизаторів протягом шести секунд. Весь цей час пакети з однаковими мітками потоку відправляються на той самий інтерфейс, що знайдений у таблиці маршрутів для першого пакету потоку.

Довжина заголовку пакета *IPv*6 дорівнює 40 байт.

Заголовок пакету *IPv*4 може мати різну довжину в залежності від наявності додаткових даних, які використовуються не часто. Ці дані потрібні для позначення таємності даних, а також під час налагоджувальних робіт щодо маршрутизації. За допомогою додаткових даних можна примусово призначати маршрути пакетів. Структуру заголовку пакетів *IPv*4 показано у таблиці 3.2 [13].

Таблиця 3.2

Структура заголовку пакета *IPv*4

Найменування параметру	Кількість біт	Значення параметру
Номер версії	4	0100 (версія 4)
Довжина заголовка у 32-бітних словах	4	від 0101 до 1111 (від 20 байт основної частини заголовка до 60 байт)
TOS Тип сервісу (*Type Of Service*)	3 1 1 1 1 1	Пріоритет від 000 до 111 (111 – вищій) 1 – мінімізувати затримку передавання 1 – максимізувати пропускну здатність 1 – максимізувати надійність 1 – мінімізувати вартість передавання 11111 – максимізувати безпечність
Загальна довжина пакета у байтах	16	Для мереж сім'ї *Ethernet* 1500 байт. Не може бути більше ніж 65500 байт
Ідентифікатор для фрагментації	16	Всі фрагменти одного пакета мають однакове значення цього ідентифікатора
Зарезервований біт Прапорець *DF* Прапорець *MF* Зміщення	1 1 1 13	Завжди нульовий 1 – заборона фрагментації пакета 1 – цей фрагмент не останній Кількість байт від початку поля даних
Час існування *TTL* (*Time To Live*)	8	Кількість вузлів, що може пройти пакет до моменту його знищення
Тип протоколу верхнього рівня	8	1 – *ICMP*, 6 – *TCP*, 17 – *UDP*
Контрольна сума заголовка	16	Цю суму перераховують на кожному вузлі після зменшення *TTL*
IP-адреса відправника пакета	32	
IP-адреса одержувача пакета	32	
Додаткові дані, яких може не бути (*IP OPTIONS*)	1 2 5	1 – слід копіювати у всіх фрагментах Клас додаткових даних (0 або 2) Номер варіанта додаткових даних
Довжина варіанта додаткових даних	8	Кількість байтів у варіанті додаткових даних
Доповнення даних кожного варіанта до 32-бітного слова	0 або 16	

Значення першого байта додаткових даних наведено у табл. 3.3.

Таблиця 3.3

Значення перших номерів опцій додаткових даних (*IP OPTIONS*)

Ознака копіювання	Клас даних	Номер опції	Значення опції
0	0	0	Кінець додаткових даних
0	0	1	Нічого не робити
1	0	2	Таємно
1	0	3	Обов'язково пройти *IP*-адреси за списком
0	2	4	Скласти список вимірів часу по вузлах
1	0	5	Цілком таємно
1	0	6	Комерційна таємниця
0	0	7	Скласти список *IP*-адрес на маршруті
1	0	8	Ідентифікатор потоку – *Stream ID*
1	0	9	Пройти тільки по *IP*-адресам, що у списку

Повний список значень опцій додаткових даних можна отримати за адресою: https://www.iana.org/assignments/ip-parameters/ip-parameters.xhtml

Додаткові дані використовують для тестування мережі. При цьому можна скористатись командою ping, у параметрах якої задають необхідний варіант додаткових даних.

Заголовок *IPv*6 спрощений у порівнянні з *IPv*4 за рахунок відсутності даних про фрагментацію та додаткових даних (*IP OPTIONS*), але всі ці можливості збережено завдяки додатковим протоколам, які показано у табл. 3.4.

Таблиця 3.4

Додаткові протоколи, які можуть доповнювати *IPv*6

Номер та назва протоколу	Призначення протоколу
0 (*Hop-By-Hop Options*)	Вказівки вузлам на шляху передавання
43 (*Routing*)	Пройти вузли за списком IP-адрес
44 (*Fragment*)	Фрагментація (розподіл на пакети обмеженої довжини)

Форму запису *IPv*6 адрес суттєво змінено у порівнянні з *IPv*4. Замість чотирьох десяткових чисел від 0 до 255, відокремлених крапками, для запису адрес *IPv*6 використовують шістнадцятирічні цифри, розділені двокрапками між

ділянками по 16 біт. Усього нараховується 8 ділянок, наприклад, *fc*00:0000:0000:0000:0000:0002:0000:*a*374.

Для скорочення запису дозволяється не писати старші нулі у кожній з ділянок, а саме цю адресу можна записати так: *fc*00:0:0:0:0:2:0:*a*374.

Також одну послідовність нульових ділянок можна замінити двома двокрапками, тоді ця ж адреса матиме скорочений вигляд: *fc*00::2:0:*a*374.

У таблицях 3.5 та 3.6 наведено переліки адрес особливого призначення для протоколів *IPv*4 та *IPv*6, відповідно.

Таблиця 3.5

Адреси *IPv*4 , що призначені для спеціального використання

Адреса	Призначення
0.0.0.0 – 0.255.255.255 (використовують лише 0.0.0.0.)	Тимчасова адреса для вузла, якому ще не надано адресу
10.0.0.0 – 10.255.255.255 172.16.0.0 – 172.31.255.255 192.168.0.0 – 192.168.255.255	Внутрішні адреси. Для адресації в межах тільки своєї (внутрішньої) мережі.
100.64.0.0 – 100.64.3.255	Розширені внутрішні мережі
127.0.0.1 – 127.255.255.255 (використовують 127.0.0.1.)	Пакет адресований своєму вузлу.
169.254.0.0 – 169.254.255.255	Мережі з автоматичною адресацією
192.88.99.9 – 192.88.99.255	Для пересилання пакетів *IPv*6
192.0.0.0 – 192.0.0.255 192.0.2.0 – 192.0.2.255 198.18.0.0 – 198.18.127.255 198.51.100.0 – 198.51.100.255 203.0.113.0 – 203.0.113.255	Зарезервовані адреси. Для тестувань та прикладів у документах
224.0.0.0 – 239.255.255.255	Групові адреси (пакети адресовані певним групам вузлів)
240.0.0.0 – 255.255.255.254	Для експериментів
255.255.255.255	На всі вузли своєї мережі

Таблиця 3.6

Адреси *IPv*6 , що призначені для спеціального використання

Адреса	Призначення
::	Аналог адреси 0.0.0.0 у *IPv*4
::1	Аналог адреси 127.0.0.1 у *IPv*4
::<адреса у форматі *IPv*4 >	Запропоновано для вузлів, які не підтримують протокол *IPv*6 (більше не використовуються)
::*ffff*:<адреса у форматі *IPv*4 >	Для вузлів, які не підтримують протокол *IPv*6
*fe*80::<ідентифікатор інтерфейсу не більше за 64 біти>	Внутрішні (*link-local*) адреси. Аналог адрес 169.254.*x.x* у *IPv*4
*fc*00::<ідентифікатор інтерфейсу не більше за 64 біти>	Внутрішні адреси. Аналог адрес типу 10.0.0.0 у *IPv*4
ffxx:< ідентифікатор групи>, де *xx* – біти ознак до адреси	Групові (*multicast*) адреси. Аналог адрес типу 224.0.0.0 у *IPv*4
2001:<ідентифікатор інтерфейсу> 2002:<ідентифікатор інтерфейсу>	Для серверів, які перетворюють адреси *IPv*4 у *IPv*6,

Головна перевага *IPv*6 над *IPv*4 полягає в тому, що назавжди зникає питання дефіциту *IP* адрес. Крім того, з’являються широкі можливості для відображення у внутрішніх адресах ознак користувачів, що спрощує задачу визначення володаря вузла по *IP* адресі.

Хоч протоколи *IPv*4 та *IPv*6 не сумісні, але засоби, що створені для полегшення переходу з *IPv*4 на *IPv*6, фактично забезпечують можливість досить довгого співіснування в мережі Інтернет обох версій протоколу. Розглянемо докладніше ці можливості.

У період даного співіснування мережа Інтернет буде складатись з ділянок, які вже отримали нові адреси довжиною 128 6іт, а решта мереж залишиться зі старими адресами по 32 біти. Розглянемо варіанти обміну даними між такими ділянками, що проілюстровано на рис. 3.1.

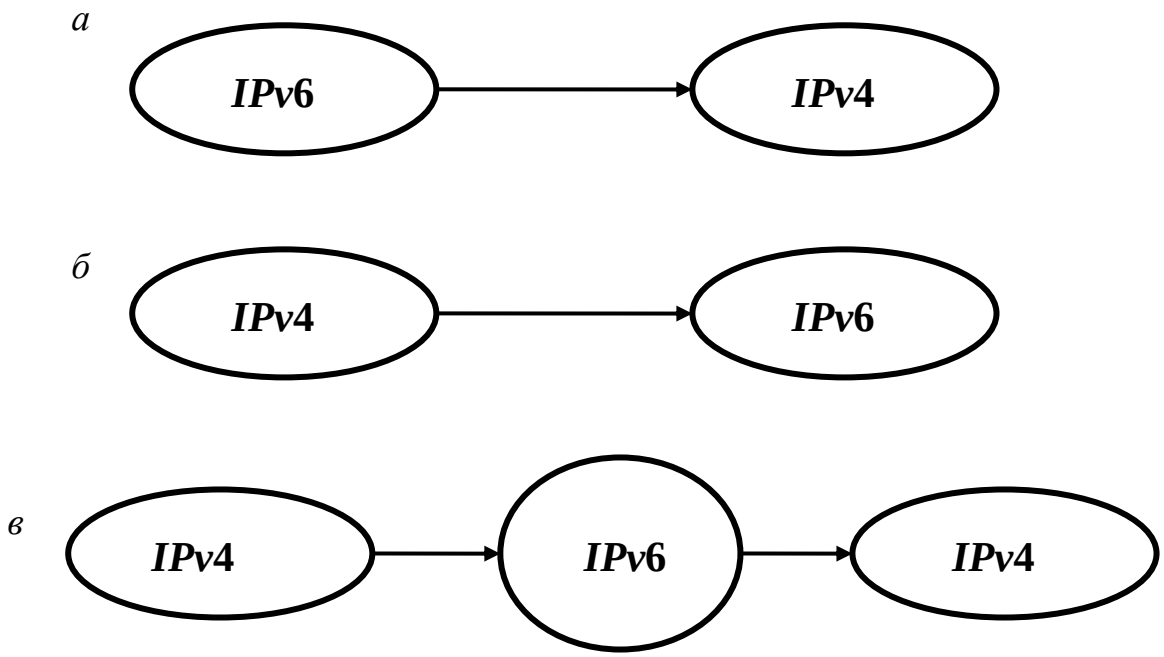


Рис.3.1. Варіанти зв'язку між ділянками з адресами *IPv*4 та *IPv*6:
а – вузол з ділянки *IPv*6 звертається до вузла з адресою *IPv*4;
б – вузол з ділянки *IPv*4 звертається до вузла зі адресою *IPv*6;
в – спілкування вузлів з адресами *IPv*4, між якими є ділянка з адресами *IPv*6

В усіх випадках між ділянками зі адресами різних типів повинні бути засоби для перетворення заголовків або для інкапсуляції пакетів. Під інкапсуляцією ми розуміємо вкладання пакету зі заголовком, що сприймає одержувач, в пакет зі заголовком, який потрібен на ділянці передавання до одержувача. Вказані засоби на рис. 3.1 позначено стрілками. Розглянемо їх особливості для кожного з варіантів.

Для першого варіанту (*а*) перетворення будуть найпростішими, бо одержувач має адресу з 32 бітів, до якої є стандартне перетворення у 128 бітну шляхом дописування на початку 80 нулів та 16 одиниць (див. у табл. 1.6 четвертий рядок). Такий пакет може пересилатись з мережі до мережі по ділянці, де діє протокол *IPv*6. На вході до ділянки одержувача потрібен сервер, який буде замінювати заголовок *IPv*6 на *IPv*4. Для цього адресу одержувача він перетворює шляхом відкидання перших 96 бітів, а 128 бітну адресу відправника він замінює на свою адресу *IPv*4. Цей сервер виконує роль посередника під час спілкування між вузлами мереж з різними типами адрес. Весь цей час він повинен підтримувати з'єднання з обома вузлами і робити пряме і зворотне перетворення заголовків різних версій протоколу *IP*.

Для другого і третього варіантів (*б, в*) проблема полягає в тому, що у пакетах *IPv*4 неможливо вмістити адресу *IPv*6. У цьому випадку пакети *IPv*6

вкладають в пакети *IPv*4 і відправляють на адресу сервера посередника, який виймає кожен пакет *IPv*6 з пакету *IPv*4 і відправляє його далі за адресою одержувача. Для цього розроблено декілька протоколів, серед яких найбільш відомі 6*to*4 [14], *6rd* [15]та *Teredo* [16].

За допомогою протоколу 6*to*4 кожний окремий комп'ютер, що має власну адресу *IPv*4 у мережі Інтернет, може стати повноцінним вузлом *IPv*6 не зважаючи на те, що мережа його провайдера не підтримує *IPv*6. Для цього на комп'ютері достатньо встановити протокол 6*to*4 з мережі Інтернет. Перед цим треба перевірити наявність зв'язку зі сервером посередником за адресою 192.88.99.1. Цим засобом підключитись до мережі *IPv*6 можна за лічені хвилини, але без гарантії якості з'єднання. Ваш комп'ютер автоматично отримає адресу *IPv*6, що матиме наступний вигляд:

2002:<32 бітна адреси вашого комп'ютера>:<80 нульових бітів>

Останні біти наданої адреси можна використовувати для створення мережі зі адресами *IPv*6, до якої доступ буде відбуватись через ваш комп'ютер. Фактично завдяки протоколу 6*to*4 в мережі *IPv*4 можна утворювати ділянки *IPv*6, які спілкуватимуться між собою крізь ділянку Інтернету, що функціонує за протоколом *IPv*4.

Протокол *6rd* (*IPv*6 *Rapid Deployment* – швидке встановлення *IPv*6) призначений для провайдерів, які ще не переобладнали свою мережу під протокол *IPv*6, але хочуть надавати клієнтам адреси *IPv*6. Перевага такого рішення у порівнянні зі 6*to*4 полягає в тому, що підключення до мережі *IPv*6 відбувається без випадкового сервера посередника, а через шлюз провайдера, де можна замовити послугу потрібної якості. При цьому мережа провайдера використовується як ділянка зі *IPv*4 для передавання пакетів *IPv*6. Адреси *IPv*6 для таких клієнтів матимуть наступний вигляд:

2001:<префікс провайдера>:<ідентифікатори мережі та інтерфейсу>.

Протокол *Teredo* дозволяє приєднатись до *IPv*6 мережі навіть тим вузлам, які не мають власної адреси, а приєднуються до Інтернету за допомогою тимчасових або внутрішніх адрес. Корпорація *Microsoft* вже встановлює в операційні системи *Windows* клієнтську частину протоколу *Teredo*. Останні доробки цього протоколу присвячені захисту клієнтських комп'ютерів від нападу зловмисників [17].

Задача протоколу *IP* полягає у доставці пакетів з даними від вузла відправника до вузла одержувача в умовах комп'ютерної мережі будь якої складності. Цю задачу він виконує не гарантую чи успішного завершення процесу передавання кожного з пакетів. Цілком можливо, що деякі пакети

будуть втрачені, але в реальних умовах таке трапляється не часто. При цьому протоколи вищих рівнів виправляють цей недолік за допомогою повторного передавання втрачених пакетів.

3.3. Протоколи мережевого рівня

Крім *IP* протоколу, який забезпечує процес передавання пакетів даних від вузла відправника до вузла одержувача, на мережевому рівні є ще протоколи маршрутизації та декілька допоміжних протоколів, які ми розглянемо в цьому розділі.

Пошук маршруту передавання пакету на кожному маршрутизаторі відбувається за допомогою таблиці маршрутів. У рядках цієї таблиці розміщено відомості про те куди (на який з інтерфейсів маршрутизатора) переслати пакет в залежності від *IP* адреси. У кожному пакеті обов'язково є *IP* адреса одержувача, яка являє собою той параметр, що необхідний для пошуку рядку у таблиці. Оскільки в мережах постійно відбуваються зміни, які найчастіше пов'язані зі появою нових вузлів, то працівникам мереж зміст таблиць маршрутів треба регулярно поновлювати. Цю роботу допомагають виконувати протоколи маршрутизації.

З точки зору маршрутизації мережа Інтернет розподілена на ділянки, які називають *AS* (*Autonomous System* – автономними системами). Кожній *AS* централізовано виділяють один або декілька блоків *IP* адрес. Кожен з цих блоків має спільний префікс (початкову частину). При цьому маршрутизація розподіляється на дві окремі задачі, одна з яких полягає в пошуку шляху до вузла в межах автономної системи, а друга – в пошуку маршруту між автономними системами, де пошук відбувається виключно за префіксами адрес. Такий підхід дозволяє скоротити кількість рядків у маршрутних таблицях і тим самим прискорити пошук маршруту.

Кожній *AS* надається номер, за яким можна визначити відповідальну особу. Це сприяє підтримці необхідного порядку для забезпечення працездатності мережі в цілому. Адміністратори *AS* повинні не тільки підтримувати маршрутизацію в межах своєї *AS*, але й забезпечити працездатність маршрутизаторів, через які відбувається зв'язок зі суміжними *AS*.

Таким чином маршрутизатори та протоколи маршрутизації чітко розподіляються на дві групи. До однієї групи належать маршрутизатори, які пересилають пакети тільки в межах *AS*, і протоколи, що мають спільну назву *IGP* (*Interior Gateway Protocol* – протокол внутрішнього шлюзу), які забезпечують

обмін інформацією про маршрути між маршрутизаторами в межах *AS*. До другої групи належать маршрутизатори, які пересилають пакети між автономними системами, і протоколи, що мають спільну назву *EGP* (*Exterior Gateway Protocol* – протокол зовнішнього шлюзу), які також забезпечують обмін маршрутною інформацією, але між маршрутизаторами другої групи.

До першої групи належать протоколи маршрутизації *RIP* (*Routing Information Protocol* – протокол маршрутної інформації) [18], *OSPF* (*Open Shortest Path First* – першим відкриває найкоротший маршрут) [19], *IGRP* (*Interior Gateway Routing Protocol* – протокол маршрутизації для внутрішнього шлюзу), та *EIGRP* (*Enhanced Interior Gateway Routing Protocol* – вдосконалений протокол маршрутизації для внутрішнього шлюзу) [20]. До другої групи належить протокол *BGP* (*Border Gateway Protocol* – протокол граничного шлюзу). Зауважимо, що шлюзами називають маршрутизатори або їх порти.

Протокол *BGP* для передавання своїх пакетів використовує протокол *TCP* транспортного рівня [21]. Через це в деяких джерелах *BGP* відносять до прикладного рівня. За прийнятими правилами протоколи нижчих рівнів надають послуги протоколам вищих рівнів, але в даному випадку маємо відхилення від цих правил. Інші протоколи маршрутизації забезпечують достовірність обміну інформацією власними засобами, а для протоколу *BGP* замість створення цих засобів скористались готовим протоколом *TCP*. Все це призначено для розв'язання задачі мережевого, а не прикладного рівня. Нагадаємо, що задачі протоколів прикладного рівня полягають у тому, щоб обслуговувати прикладні процеси, а не процедури маршрутизації.

Користь від протоколів маршрутизації полягає в тому, що тільки в одну таблицю повинен адміністратор заносити дані. Протоколи маршрутизації ці дані автоматично перенесуть до інших маршрутизаторів, яким необхідно мати відомості про даний маршрут. У разі коли кількість маршрутизаторів у мережі не перевищує 3-4, то використання протоколів маршрутизації вважається недоцільним.

У кожному рядку таблиці маршрутів знаходяться наступні дані:

– префікс адреси мережі, до якої веде цей маршрут (для протоколу *IPv*4 замість префіксу іноді надають адресу та маску мережі);
– інтерфейс, на який слід відправляти пакети за цим маршрутом;
– адреса інтерфейсу наступного вузла на цьому маршруті;
– метрика маршруту (числове значення, яке використовують для вибору маршруту у разі наявності декількох можливих маршрутів).

Перелічені дані необхідні для пошуку інтерфейсу, на який слід відправити пакет. Крім цього, в залежності від протоколу обміну маршрутною інформацією, у таблиці можуть бути розміщені наступні дані, що необхідні для коригування змісту самих таблиць:

– залишок часу існування маршруту (кількість секунд до знищення даного рядку таблиці);
– ознака необхідності відправлення інформації про цей маршрут до інших маршрутизаторів;
– проміжок часу після поновлення даних про цей маршрут;
– ознака активності маршруту.

Кожен порт кожного маршрутизатора має свою *IP* адресу, що потрібно для пересилання пакетів між маршрутизаторами.

Розглянемо процедуру пошуку маршруту на прикладі фрагменту університетської мережі (КНУБА), що зображений на рис. 3.2.

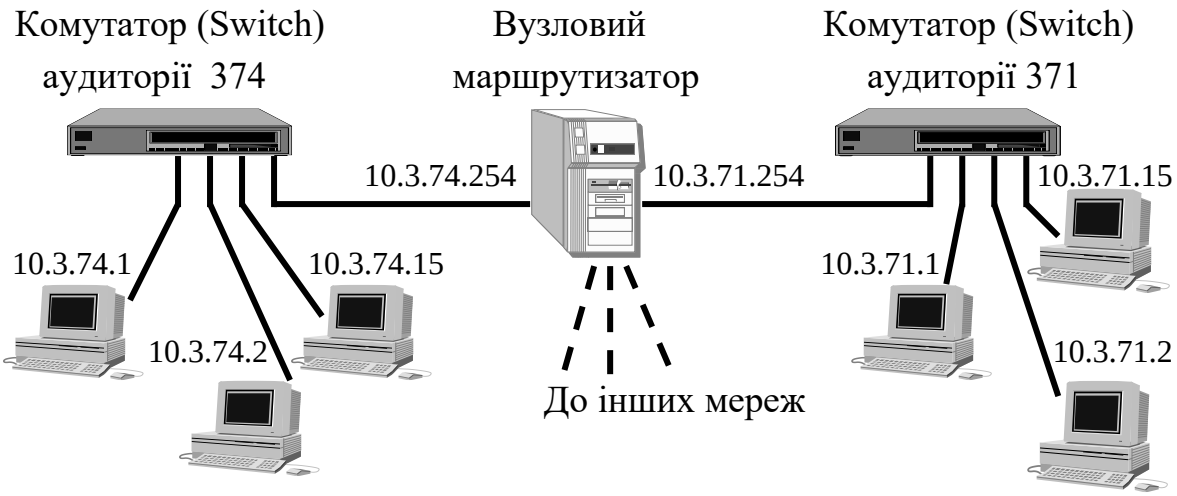


Рис. 3.2. Схема мережі для пояснення процедури маршрутизації

В мережах аудиторій 371 та 374 встановлено по 15 комп'ютерів (з них на рисунку показано по троє). Вузловий маршрутизатор являє собою програмний засіб на окремому комп'ютері під операційною системою *Free BSD*. Маршрутизатори (також у вигляді програмних засобів) є у кожному комп'ютері. Розглянемо блок-схему мережевого програмного забезпечення кінцевого вузла мережі на прикладі комп'ютера з адресою 10.3.74.1, що зображена на рис. 3.3.

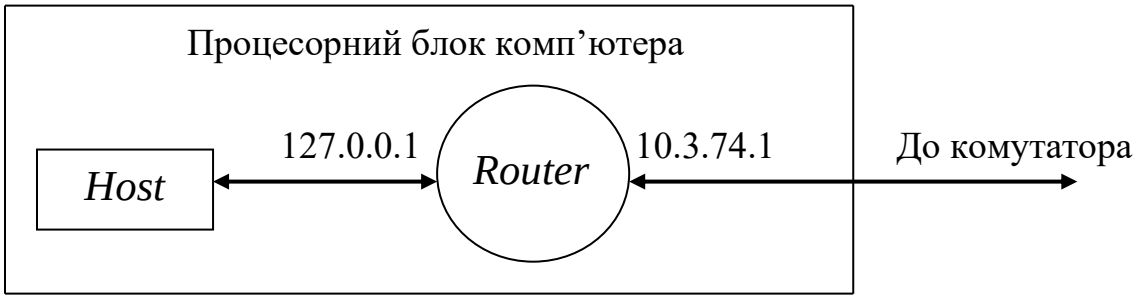


Рис. 3.3. Схема мережевого програмного забезпечення комп'ютера з адресою 10.3.74.1

Надамо роз'яснення про те для чого потрібен маршрутизатор (*Router*) у кожному кінцевому вузлі (*Host*) мережі. Для цього подивимось на таблицю маршрутів даного маршрутизатора, яку в операційний системі *Windows* можна роздрукувати за допомогою команд *netstat –r* або *route print*. Цю таблицю зображено на рис. 3.4.

Активні маршрути:

Адреса мережі	Маска мережі	Адреса шлюзу	Інтерфейс	Метрика
0.0.0.0	0.0.0.0	10.3.74.254	10.3.74.1	1
127.0.0.0	255.0.0.0	On-link	127.0.0.1	1
10.3.74.0	255.255.255.0	On-link	10.3.74.1	1
10.3.74.1	255.255.255.255	On-link	127.0.0.1	1
10.3.74.255	255.255.255.255	On-link	10.3.74.1	1
224.0.0.0	240.0.0.0	On-link	10.3.74.1	1
255.255.255.255	255.255.255.255	On-link	10.3.74.1	1

Рис. 3.4. Таблиця маршрутів у комп'ютері з адресою 10.3.74.1

Дана таблиця відповідає четвертій версії *IP* протоколу, де довжина адрес дорівнює 32 бітам. Такі адреси записують чотирма десятковими числами від 0 до 255. Кожному числу відповідає 8 бітів (один байт). Перші два стовпчики таблиці дозволяють визначити префікс адреси за допомогою побітової операції *AND* між адресою і маскою. Замість маски мережі у таблицях *IPv6* після адреси вказують кількість бітів префіксу. В деяких системах такий запис застосовують також і з адресами *IPv4*, наприклад, 10.3.74.0/24 означає, що довжина префіксу становить 24 біти. Це дорівнює значенню маски 255.255.255.0, яка у бітах має вигляд:

11111111 11111111 11111111 00000000

Кількість одиниць у масці означає довжину префіксу. Комп'ютери однієї мережі мають адреси з однаковим префіксом. Біти після префіксу використовують для адресації комп'ютерів в даній мережі. Накладання маски на

IP адресу за допомогою операції *AND* фактично виділяє префікс шляхом заміни решти бітів на нулі.

Розглянемо процедуру визначення належності комп'ютера до мережі на прикладі третього рядку таблиці (див. рис. 3.4). У цьому рядку адреса мережі дорівнює 10.3.74.0. Якщо на адресу комп'ютера 10.3.74.2 накласти маску цієї мережі 255.255.255.0, то одержимо 10.3.74.0, що співпадає з адресою мережі. Це означає належність комп'ютера до даної мережі. Легко побачити, що комп'ютер з адресою 10.3.71.1 не належить до даної мережі. Належність комп'ютера до мережі простіше отримати порівнянням префіксів адреси мережі і адреси комп'ютера.

По двох маршрутах даної таблиці (другий та четвертий рядки) пакети слід відправляти на інтерфейс зі адресою 127.0.0.1, що означає їх повернення до відправника. Цю адресу зарезервовано для тестування програмного забезпечення стеку *TCP/IP*. Навіть якщо комп'ютер не підключено до жодної мережі, то пакети можна пересилати на цю адресу. Більше 16 мільйонів адрес з префіксом 127.0.0.0/8 (від 127.0.0.0 до 127.255.255.255) зарезервовані для такого тестування і не можуть використовуватись в інших цілях. Пакети на всі ці адреси будуть відправлені за маршрутом другого рядку таблиці на інтерфейс 127.0.0.1. Цей маршрут має назву *loop* (петля). У версії протоколу *IPv6* для цього призначено тільки одну адресу ::1. У четвертому рядку таблиці наведено маршрут для пакетів, які відправляють на власну адресу 10.3.74.1. Тут теж маємо інтерфейс 127.0.0.1. У цьому рядку розглядається мережа з одного комп'ютера, бо довжина префіксу дорівнює довжині адреси.

Заради розглянутих двох маршрутів і знадобилися маршрутизатори на кожному кінцевому вузлі мережі.

Три останні рядки таблиці відповідають адресам типу *broadcast* (широкомовна) та групова. Групові адреси ми розглянемо далі із протоколом *IGMP*.

Для маршрутів, де у стовпчику «Адреса шлюзу» вказано *On-link*, вузол одержувача знаходиться у тій самій мережі, що й цей комп'ютер. В деяких таблицях замість слів *On-link* дублюється адреса інтерфейсу.

Пакети з цього комп'ютера можуть пересилатись як в межах аудиторії, так і в інші мережі через наступний маршрутизатор. У цьому разі пакети будуть відправлені за маршрутом, який вказано у першому рядку таблиці, що зветься маршрутом по замовчанню. Його позначають адресою з усіх нулів і нульовим префіксом. Цей маршрут використовують тоді, коли інших маршрутів не

знайдено. Такі пакети відправляють на адресу шлюзу, що є портом наступного маршрутизатора, який пересилатиме їх до інших мереж.

На першому етапі маршрутизації відбувається пошук маршруту у таблиці відправника пакету. Другий етап може мати три наступні варіанти.

- Пакет слід повернути відправнику на інтерфейс 127.0.0.1.
- Пакет слід відправити одержувачу в межах даної мережі.
- Пакет слід переслати до іншої мережі.

У першому варіанті пакет пересилається в межах комп'ютера без використання технологій канального рівня.

У другому і третьому варіантах необхідно пересилати пакет за допомогою обладнання канального рівня. Для цього слід сформувати кадр зі структурою, яка зображена на рис. 1.10 або рис. 2.14.

В нашому випадку канальний рівень побудовано за технологією *Fast Ethernet* (див. підрозділ 2.2.3). Канальне обладнання сім'ї *Ethernet* автоматично обирає тип кадрів і найкращий режим роботи. Єдиний параметр, який треба знайти, це фізична адреса (*MAC*-адреса) одержувача. Цей пошук для випадку *IPv*4 виконує протокол мережевого рівня *ARP* (*Address Resolution Protocol* – протокол, що знаходить фізичну адресу відповідну до мережевої). Пошук адреси відбувається за допомогою таблиці, яку можна роздрукувати за допомогою команди *arp -a* у всіх операційних системах. Вигляд цієї таблиці представлено на рис. 3.5.

```
Інтерфейс: 10.3.74.1
  IP-адреса      Фізична адреса        Тип
  10.3.74.3      00:13:4A:25:78:E7     Динамічна
  10.3.74.254    00:13:4A:25:78:B1     Статична
```

Рис. 3.5. Таблиця відповідності адрес за протоколом *ARP*

Значення типу у цій таблиці вказує на те яким чином сформовано даний рядок. Динамічні рядки заповнюються автоматично на обмежений термін (найчастіше на 2 хвилини) за допомогою процедури запитів та відповідей, яка є складовою протоколу *ARP* (документ *RFC* 826). Статичні рядки залишаються незмінними. Їх можна заповнювати вручну з метою захисту мережі від зловмисників, але цим засобом користуються не часто, бо автоматичні засоби працюють досить надійно і майже не витрачають мережеві ресурси. Для отримання фізичної адреси в автоматичному режимі з боку відправника повідомлення формується пакет-запит за стандартною формою. Ця форма є

єдиною для запитів і відповідей (табл. 3.7). Як запити, так і відповіді пересилаються безпосередньо в кадрах *Ethernet* зі значенням параметру *Type* = 2054. Запити відправляють широкомовно на адресу *FF-FF-FF-FF-FF-FF*. Відповідь на запит надає лише той вузол, в якого *IP*-адреса співпадає з адресою одержувача пакета *ARP*. Після отримання відповіді заповнюється рядок динамічного типу у таблиці протоколу *ARP*. Якщо відповідь не отримано, то маршрутизатор інформує відправника про недосяжність одержувача, формуючи повідомлення за протоколом *ICMP* (*Internet Control Message Protocol* — протокол діагностичних повідомлень та управління зв'язком між мережами), який ми розглянемо нижче.

Таблиця 3.7

Структура *ARP*-пакета у мережі *Ethernet* зі стеком *TCP/IP* (*IPv*4)

Найменування параметру	Кількість біт	Значення параметру
Код технології канального рівня	16	1 для *Ethernet*
Код протоколу мережевого рівня	16	2048 для протоколу *IP*
Довжина фізичної адреси	8	6 (байт у *MAC*-адресі)
Довжина мережевої адреси	8	4 (байти у *IP*-адресі)
Код операції *ARP* (запит/відповідь)	16	1- запит, 2 - відповідь
MAC-адреса відправника пакета *ARP*	48	
IP-адреса відправника пакета *ARP*	32	
MAC-адреса одержувача пакета *ARP*	48	Нулі у запиті
IP-адреса одержувача пакета *ARP*	32	

Для *IPv*6 протокол *ARP* не використовується. Його функції виконує поновлений протокол *ICMP*, що має назву *ICMPv*6.

Головна різниця у формуванні запитів для випадків коли одержувач знаходиться в даній мережі (*On-link*) і тоді коли одержувач поза межами даної мережі полягає в тому, що у першому випадку запит фізичної адреси робиться за *IP* адресою одержувача, а у другому – за *IP* адресою шлюзу. Після отримання фізичної адреси *IP* пакет розміщується в кадрі *Ethernet* зі значенням параметру *Type* = 2048 для *IPv*4 або *Type* = *Ox*86*DD* для *IPv*6 і відправляється на наступний маршрутизатор.

В першому випадку наступний маршрутизатор знаходиться в комп'ютері одержувача. Його таблиця подібна до тієї, яку зображено на рис. 3.4, але замість адреси 10.3.74.1 там буде адреса одержувача. Маршрут, за яким пакет буде

відправлено на свій комп'ютер за адресою 127.0.0.1, є у четвертому рядку таблиці.

У випадку коли одержувач знаходиться поза межами даної мережі, пакет буде відправлено по замовчанню на вузловий маршрутизатор (див. рис. 3.2) за адресою 10.3.74.254. Фрагмент його таблиці для маршрутів, які ми розглядаємо, представлено у табл. 3.8.

Таблиця 3.8

Основні параметри маршрутів вузлового маршрутизатора

Префікс адреси мережі	Адреса шлюзу	Адреса інтерейсу
0.0.0.0/0	192.168.1.1	192.168.1.100
10.3.71.0/24	On-link	10.3.71.254
10.3.74.0/24	On-link	10.3.74.254

У випадку передачі даних між аудиторіями з адреси 10.3.74.1 на адресу 10.3.71.3 вузловий маршрутизатор, отримавши *IP* пакет, починає пошук маршруту у своїй таблиці (див. табл. 3.8). Знайшовши потрібний маршрут шляхом порівняння префіксів (другий рядок у табл. 3.8), він відправляє широкомовний запит фізичної адреси з інтерфейсу 10.3.71.254 на всі комп'ютери мережі аудиторії 371. Отримавши відповідь від комп'ютера 10.3.71.3 з його фізичною адресою, вузловий маршрутизатор відправляє *IP* пакет в кадрі *Ethernet* на комп'ютер одержувача.

Крім пошуку маршрутів і пересилання *IP* пакетів маршрутизатори виконують ряд дій по перевірці кожного пакета. У разі виявлення помилки маршрутизатор знищує *IP* пакет та відправляє повідомлення на вузол відправника зі роз'ясненням причини цього знищення. Такі повідомлення формуються за протоколом *ICMP* у випадку використання *IPv*4 або за протоколом *ICMPv*6 для *IPv*6.

Протокол *ICMP* для передавання своїх повідомлень використовує протокол *IP*, що наближає його до транспортного рівня, але він не виконує задач транспортного рівня, а тільки обслуговує процедуру доставки *IP*-пакетів, тому протокол *ICMP* відносять до мережевого рівня. У *ICMPv*6 об'єднано функції трьох попередніх протоколів, а саме *ARP*, *ICMP* та *IGMP* (Internet Group Management Protocol – протокол управління груповою адресацією в Інтернеті). Зауважимо, що групова адреса може надаватись інтерфейсам вузлів як додаткова. Індивідуальна адреса є необхідною для кожного інтерфейсу, що підключений до мережі, бо адреса відправника може бути тільки індивідуальною і без неї вузол не зможе відправити *IP*-пакет. Групові адреси використовують для

обміну маршрутною інформацією між маршрутизаторами. Наприклад, в межах автономної системи множині маршрутизаторів надають єдину групову адресу. При цьому кожен відправлений пакет за такою адресою потрапляє на всі ці маршрутизатори. Таким чином зменшується кількість відправлень пакетів під час доставки однакової інформації групам одержувачів.

Протокол *IGMP* дозволяє автоматично виявляти множини вузлів, яким надаються групові адреси, на кожному з інтерфейсів мережі, та підтримувати в актуальному стані дані про маршрути до цих вузлів у маршрутних таблицях. Структуру пакетів протоколу *IGMP* показано у табл. 3.9.

Таблиця 3.9

Структура *IGMP*-пакета

Найменування параметру	Кількість біт	Значення параметру
Тип повідомлення	8	17 – запит, 18 або 22 – відповідь, 23 – вихід з групи
Максимальний час очікування відповіді	8	У частках по 0,1 секунди (тільки у запитах)
Контрольна сума	16	
Групова *IP*-адреса	32	Нулі у загальних запитах

Робота даного протоколу розпочинається зі загального запиту, що відправляється до всіх вузлів своєї мережі за спеціальною груповою адресою 224.0.0.1, яка призначена виключно для цих запитів. На цей запит вузол опитувач протягом встановленого проміжку часу (по замовчанню 10 секунд) повинен отримати відповіді від тих вузлів, які приймають або пересилають пакети за груповою адресою (крім адреси 224.0.0.1). Кількість відповідей від кожного з вузлів дорівнює кількості відомих їм групових адрес. Отримавши відповіді вузол опитувач заповнює власну маршрутну таблицю даними щодо маршрутизації за груповими адресами. Загальні запити відправляються вузлами з інтервалом у 125 секунд. Цей інтервал у разі необхідності можна змінити.

Зауважимо, що впровадження протоколу *IPv*6 в мережі Інтернет неможливо реалізувати за короткий час. Період співіснування вузлів з адресами *IPv*4 та *IPv*6 може розтягнутись на довгі роки. Тому розглянемо обидва варіанти протоколу *ICMP* для *IPv*4 та *IPv*6.

Усі версії протоколу *ICMP* мають однакову структуру пакетів, яку зображено у таблиці 3.10.

Таблиця 3.10

Структура пакета *ICMP*

Найменування параметру	Кількість біт	Значення параметру
Тип повідомлення (*Type*)	8	Для версії *ICMP*, що співпрацює зі *IPv*4, ці значення надано у таблиці 2.14, а для версії *ICMPv*6 – у таблиці 2.15
Код повідомлення (*Code*)	8	В залежності від типу повідомлення уточнює причину знищення пакета
Контрольна сума	16	
Додаткові дані сталої довжини	32	Використовуються у деяких типах повідомлень або заповнюються нулями
Додаткові дані змінної довжини		Найчастіше початкові 28 байт *IP*-пакету, що був знищений

Значення типів та кодів для повідомлень наведено у табл. 3.11.

Таблиця 3.11

Значення типів та кодів повідомлень *ICMP* для *IPv*4

Тип	Код	Значення повідомлення
0	0	Луна-відповідь
3	0	Мережа недосяжна
	1	Вузол недосяжний
	2	Протокол верхнього рівня не підтримується
	3	Порт недосяжний
	4	Необхідна фрагментація, але є прапорець заборони
	5	Хибний маршрут від відправника
	6	Невідома адреса мережі
	7	Невідома адреса вузла
	8	Вузол ізольовано

Закінчення табл. 3.11

Значення типів та кодів повідомлень *ICMP* для *IPv4*

Тип	Код	Значення повідомлення
3	9	Адміністративна заборона доступу до мережі
	10	Адміністративна заборона доступу до вузла
	11	Мережа недосяжна для даного типу сервісу
	12	Вузол недосяжний для даного типу сервісу
4	0	Заборона передавання через перевантаження маршруту
5	0	Направлення пакета у мережу
	1	Направлення пакета на вузол
	2	Направлення пакета у мережу за типом сервісу
	3	Направлення пакета на вузол за типом сервісу
8	0	Луна-запит
11	0	Вичерпано час існування пакета
	1	Вичерпано час збирання фрагментів
12	0	У додаткових даних зазначено номер хибного байта
	1	Відсутній необхідний варіант додаткових даних у *IP*-заголовку (це може бути варіант 2 – таємно або 5 – цілком таємно)
	2	Хибне значення довжини пакета

Повідомлення типів 8 та 0 використовуються для перевірки зв’язку між вузлами мережі. На цих повідомленнях побудовано широко відому команду *ping*, яка є зручним інструментом для тестування мереж. При цьому у додаткових даних сталої довжини розміщуються ідентифікатор (перші 16 біт) та номер запиту (останні 16 біт). Значення ідентифікатора однакове впродовж сеансу тестування, а номер запиту являє собою послідовний номер (починаючи з нуля) спроб обміну пакетами в межах сеансу тестування. У пакетах луна-відповідь номери співпадають зі відповідними пакетами луна-запит.

Більшість варіантів повідомлень *ICMP* призначені для інформування відправника про аварійні ситуації, але не в усіх випадках аварій слід відправляти такі повідомлення. Не відправляють *ICMP*-повідомлення про знищення *IP*-пакетів зі широкомовними та груповими адресами, фрагментів *IP*-пакетів, крім першого, та про знищення пакетів *ICMP*. Також заборонено відправляти повідомлення у разі знищення пакетів через перевантаження маршруту, хоч для таких повідомлень визначено тип 4.

Для версії *ICMPv*6 значення типів та кодів повідомлень наведено у табл. 3.12.

Таблиця 3.12

Значення типів та кодів повідомлень *ICMPv*6

Тип	Код	Значення повідомлення
1	0	Відсутній маршрут до одержувача
	1	Адміністративна заборона доступу до одержувача
	2	Адреса відправника за межами існуючих
	3	Адреса одержувача недосяжна
	4	Порт одержувача недосяжний
	5	Адреса відправника заборонена політикою безпеки
	6	Маршрут до одержувача ліквідовано
	7	Помилковий маршрут від відправника (*RFC* 4443)
2	0	Пакет занадто довгий (*RFC* 4443)
3	0	Вичерпано час існування пакета
	1	Вичерпано час збирання фрагментів (*RFC* 4443)
4	0	Помилковий параметр у заголовку
	1	Невідомий тип вкладеного протоколу
	2	Невідома опція *IPv*6 (*RFC* 4443)
128	0	Луна-запит (*RFC* 4443)
129	0	Луна-відповідь (*RFC* 4443)
130	0	Запит для виявлення вузлів, що сприймають групові (*multicast*) адреси (*RFC* 2710)
131	0	Відповідь вузла, що сприймає групову адресу
132	0	Відповідь вузла, що припиняє сприймати групову адресу
133	0	Запит для виявлення маршрутизаторів (*RFC* 4861)
134	0	Відповідь маршрутизатора
135	0	Запит для виявлення сусідніх вузлів
136	0	Відповідь сусіднього вузла
137	0	Направлення на кращий маршрут (*RFC* 4861)
138	0	Команда зміни адреси маршрутизатора
	1	Відповідь про зміну адреси
	255	Відмова від зміни адреси
139	0	Запит інформації вузла за адресою *IPv*6
	1	Запит інформації вузла за ім’ям комп’ютера
	2	Запит інформації вузла за адресою *IPv*4 (*RFC* 4620)
140	0	Відповідь на запит інформації вузла
	1	Відмова у наданні інформації
	2	Відповідь про відсутність інформації (*RFC* 4620)
141	0	Запит вузла для одержання *IP*-адреси (*RFC* 3122)
142	0	Відповідь на запит вузла для одержання *IP*-адреси
143	0	Відповідь вузла, що сприймає групову адресу і працює зі протоколом *MLDv*2 (*RFC* 3810)
144	0	Запит з мобільного вузла, який має *IP*-адресу (*RFC* 6275)
145	0	Відповідь мобільному вузлу, який має *IP*-адресу

Закінчення табл. 3.12

Значення типів та кодів повідомлень *ICMPv*6

Тип	Код	Значення повідомлення
146	0	Запит префіксу адреси мобільного вузла (*RFC* 6275)
147	0	Оповіщення про префікс адреси мобільного вузла
148	0	Запит сертифікату для захищеного зв'язку (*RFC* 3971)
149	0	Оповіщення про сертифікат для захищеного зв'язку
150	0	Повідомлення для експериментальних протоколів мобільного зв'язку (*RFC* 4065)
151	0	Запит для виявлення маршрутизаторів, що сприймають групові (*multicast*) адреси (*RFC* 4286)
152	0	Відповідь маршрутизатора, що сприймає групову адресу
153	0	Відповідь маршрутизатора, що припиняє сприймати групову адресу
154	0-5	Запит до маршрутизаторів з метою виявлення наступного посередника для мобільного вузла під час руху та оповіщення від маршрутизаторів (*RFC* 5568)
155	0-3, 16-19, 26	Запити та відповіді вузлів малої потужності, які працюють за алгоритмом однорангових (*p*2*p*) мереж (*RFC* 6550)

Кількість повідомлень протоколу *ICMPv*6 збільшено порівняно із *ICMP* за рахунок доповнення кодів 130-132, які замінюють функції протоколу *IGMP*, але при цьому введено протокол *MLD* (*Multicast Listener Discovery*). Додано коди для поліпшення процедури маршрутизації в умовах роботи з мобільними вузлами. Але аналіз темпів впровадження *IPv*6, показує, що на червень 2022 року (10 років з початку переведення магістральних мереж Інтернету на версію *IPv*6) лише 28% ресурсів Інтернету підтримують нову версію [22]. Широко відома компанія *Microsoft* до цього часу залишає свої сервери працювати тільки з *IPv*4. Суттєвого покращення характеристик роботи мережі Інтернет від переходу на нову версію протоколу *IP* не було виявлено і процес впровадження *IPv*6 протягом перших десяти років відбувався повільно. За цей час трафік у мережі збільшився у 50 разів.

3.4. Протоколи транспортного рівня

Головна задача транспортного рівня полягає у створенні множини з'єднань через один мережевий інтерфейс.

Кожен вузол мережі може одночасно підтримувати більше ніж одне мережеве з'єднання (буває, що число цих з'єднань досягає десятків тисяч). При цьому *IP* адреса вузла найчастіше є тільки одна. Для того, щоб в межах комп'ютера відрізняти пакети кожної з діючих програм, адреси доповнюють номерами, які називають портами транспортного рівня. В заголовках усіх протоколів транспортного рівня на першому місці вказують значення портів відправника (*Source port*) та одержувача (*Destined port*) пакету, на кожний з них виділено по 16 бітів. Це означає, що загальна кількість номерів портів для різних процесів на кожному вузлі мережі для кожного з напрямків передачі може сягати більше ніж 65 тисяч. Таким чином транспортний рівень забезпечує можливість на кожному вузлі створити потрібну кількість зв'язків між діючими програмами на своєму і віддаленими вузлами мережі.

Переважна більшість сеансів зв'язку в мережі Інтернет відбувається за протоколом *TCP*, який забезпечує встановлення наскрізного прозорого сполучення між вузлами. Функції протоколу *TCP* полягають у наступному.

- Призначення номерів портів відправнику і одержувачу даних (до 65536 портів кожному з них).
- Перевірка зв'язку та встановлення з'єднання між відправником та одержувачем даних. У разі невдалої спроби з'єднання на верхній (прикладний) рівень видається сигнал про аварію.
- Вибір оптимальної швидкості передавання пакетів із врахуванням можливостей одержувача і каналу зв'язку.
- Перевірка вірності передавання кожного пакета за допомогою контрольної суми. Хибні пакети знищуються, а на вірні пакети відправляється підтвердження. У разі затримки підтвердження пакет передається повторно.
- Контроль послідовності надходження байтів. Для цього усі байти нумерують. У разі виявлення порушення послідовності байтів їх впорядковують.
- Припинення передавання та розірвання з'єднання у разі затримки пакетів на час, що перевищує визначений тайм-аут. При цьому прикладні процеси відправника та одержувача інформуються про аварію.

- Розірвання з’єднання у разі завершення передавання даних. При цьому прикладні процеси відправника та одержувача інформують про те, що усі дані передано.

Структуру заголовка протоколу *TCP* наведено у таблиці 3.13.

Таблиця 3.13

Структура *TCP*-заголовка

Найменування параметру	Кількість біт	Значення параметру
Порт відправника (*Source port*)	16	Номер з’єднання формується автоматично
Порт одержувача (*Destined port*)	16	Номер означає протокол прикладного рівня (23 – *TELNET*, 25 – *SMTP*, 80 – *HTTP*, 110 – *POP3*, 21 – *FTP*)
Номер байта даних, що передається першим у пакеті	32	За початок відліку беруть випадкове число
Номер байта даних, що очікується	32	Цей номер може залишатись незмінним у разі відсутності інформації, крім сигналів-квитанцій (підтверджень)
Кількість 32-бітних слів у *TCP*-заголовку	6	5 або 6
Не використовують	4	Завжди нульові
URG (*Urgent*), що означає терміновість	1	1 – є термінові дані 0 – немає термінових даних
ACK (*Acknowledge*) – підтвердження	1	1 – пакет прийнято без помилок 0 – це початковий пакет сеансу зв’язку
PSH (*Push*) – виштовхування	1	1 – дані слід передати одержувачу не очікуючи наступний пакет
RST (*Reset*) – відмова	1	1 – аварійне припинення зв’язку
SYN (*Sync*) – синхронізація	1	1 – у перших двох пакетах сеансу зв’язку
FIN (*Final*) – кінець	1	1 – у завершальних трьох пакетах
Розмір вікна	16	Кількість вільних байт у буфері
Контрольна сума	16	Сумуються усі дані сегменту та частина заголовку *IP* (псевдо заголовок)
Кількість термінових байт у пакеті	16	
MSS (*Max segment size*) – максимальний розмір сегмента	16	Максимальна кількість байт у сегменті (передають тільки на початку сеансу)
Доповнення до 32-бітного слова	16	Використовують тільки до даних про максимальний розмір сегмента

Початок *TCP*-заголовка займають номери портів, що являють собою доповнення до *IP*-адрес відправника та одержувача.

Термін порт у комп'ютерній лексиці часто використовують як назву для тієї чи іншої фізичної або логічної точки доступу.

Для доставки пакета зі точністю до комп'ютера достатньо знати IP-адресу, але в комп'ютері можуть одночасно підтримуватись у активному стані багато процесів зв'язку і необхідно визначати до якого з цих процесів належить кожний пакет. Наприклад, коли ми відкриваємо декілька вікон до якого-небудь сайту то усі пакети мають абсолютно однакові IP-адреси, але протокол *TCP* надає їм різні номери портів.

На рис.3.14 відображено значення даних у *TCP*-заголовках для трьох початкових пакетів сеансу зв'язку.

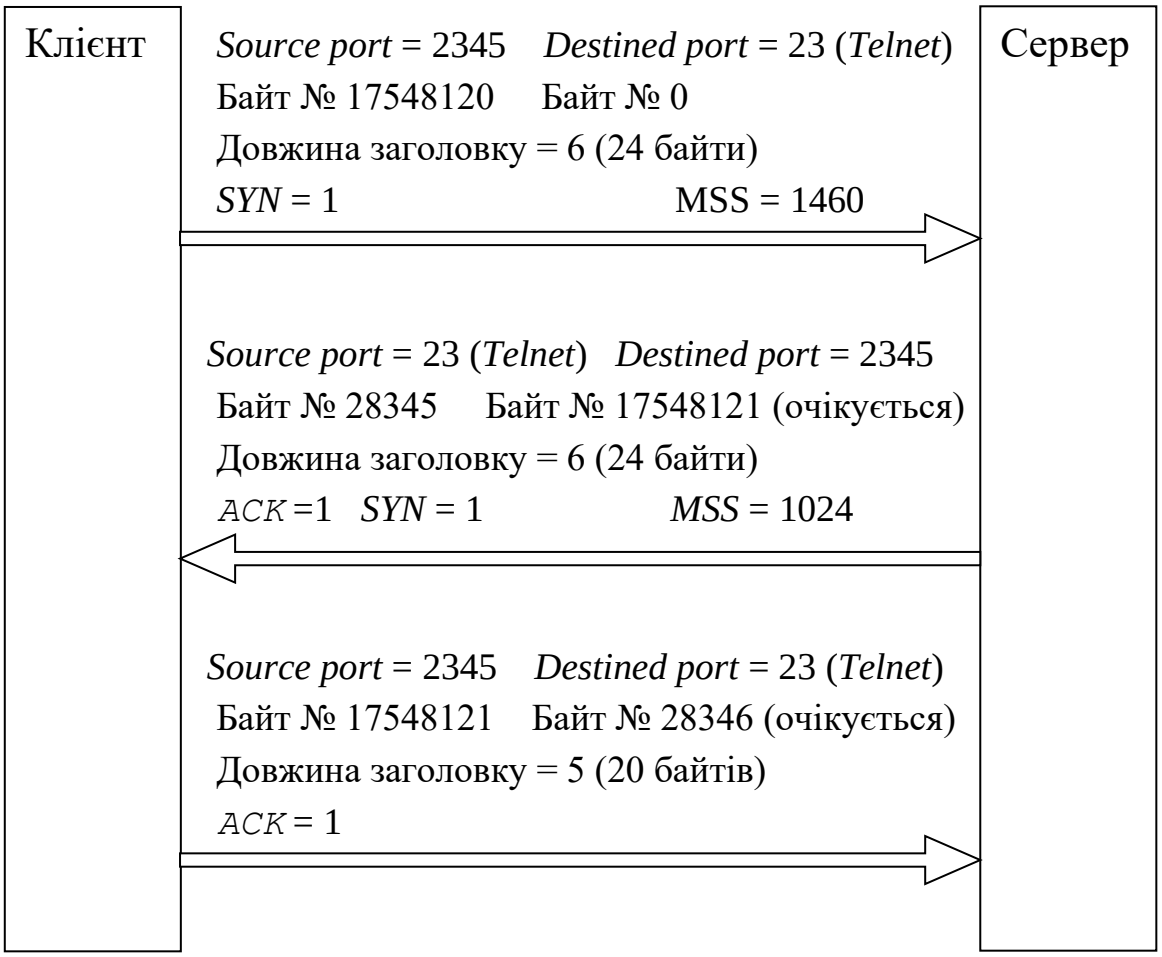


Рис. 3.14. Процедура встановлення з'єднання за протоколом *TCP*

Після успішного обміну першими трьома пакетами, що називають процедурою встановлення з'єднання, розпочинається обмін даними. У нашому випадку, коли протоколом верхнього рівня є *Telnet*, з боку клієнта відправляються команди з клавіатури, а у відповідях приходять дані для

відображення на екрані терміналу. Цей режим обміну даними широко розповсюджений у роботі адміністраторів вузлів мережі Інтернет.

Крім протоколу *TCP* на транспортному рівні стеку *TCP/IP* є інші протоколи, серед яких важливе місце займає протокол *UDP*, структуру заголовку якого надано у таблиці 1.14. Цей протокол не гарантує надійної доставки даних, але забезпечує мінімальну затримку передавання. Переваги цього протоколу обумовлені можливістю передавання широкомовних та термінових повідомлень.

Слід зауважити, що не для кожного сеансу зв'язку є необхідність у абсолютно надійній доставці пакетів. Наприклад, у разі телевізійної або радіо трансляції знищення деяких пакетів майже не впливає на якість передавання. Це у значній мірі стосується також Інтернет телефонії. Тут використання протоколу *TCP* може призвести до негативних наслідків, бо у разі виявлення помилки почнеться повторна передача пошкодженого пакету, а це викличе затримку або довгу паузу, що значно погіршує якість сприйняття інформації у реальному часі.

Для подібних задач створено низку протоколів транспортного рівня, серед яких найбільш відомими є *SCTP* (*Stream Control Transmission Protocol* – протокол передачі з управлінням потоком) та *DCCP* (*Datagram Congestion Control Protocol* – протокол управління заторами пакетів). Ці протоколи призначені для покращання якості зв'язку в умовах реального часу і перш за все Інтернет телефонії. Протокол *SCTP* має вдосконалену процедуру встановлення з'єднань, яка на відміну від протоколу *TCP* надає захист від атак типу *DoS* (відмова в обслуговуванні), які унеможливлюють доступ до мережевих ресурсів. Впровадження нових протоколів потребує досить довгих випробувань, що у кожному випадку займає близько десяти років. Весь цей час характеристики протоколів відкрито обговорюються і вдосконалюються. У разі наявності дійсно корисного ефекту протоколи стандартизуються і впроваджуються. У цьому процесі можуть прийняти участь усі бажаючі фахівці. Все це легко відстежити ознайомлюючись з документами *RFC* на сайті *http*: //*tools.ietf.org*.

В и с н о в к и

1. Потреба в забезпеченні зв'язку між комп'ютерами, що належать різним мережам, без обмежень на їх територіальне розміщення і кількість, призвела до створення протоколів мережевого та транспортного рівнів. В результаті було отримано об'єднану мережу, яка складається з необмеженої кількості мереж.

2. Для адресації вузлів в об’єднаній мережі використовують *IP* (ай-пі) адреси, які надаються за протоколом мережевого рівня *IP* (*Internet Protocol*). Загальна кількість адрес для четвертої версії *IP* протоколу *IPv*4 перевищує 4 мільярди (2^{32}), що до червня 2012 року ще вистачало користувачам Інтернету, але з 6 червня 2012 року майже всі магістральні мережі переведено на шосту версію *IPv*6, у якій кількість адрес практично необмежена (2^{128}).
3. Протоколи *IPv*4 та *IPv*6 не сумісні, але засоби, що створені для полегшення переходу з *IPv*4 на *IPv*6, фактично забезпечують можливість досить довгого співіснування в мережі Інтернет обох версій протоколу.
4. Форму запису *IPv*6 адрес суттєво змінено у порівнянні з *IPv*4. Замість чотирьох десяткових чисел від 0 до 255, відокремлених крапками, для запису адрес *IPv*6 використовують шістнадцятирічні цифри, розділені двокрапками між ділянками по 16 біт. Усього нараховується 8 ділянок. Для скорочення запису дозволяється не писати старші нулі у кожній з ділянок. Одну послідовність нульових ділянок можна замінити двома двокрапками.
5. Крім надання унікальних адрес вузлам складної мережі необхідно ще відшукувати шлях (маршрут) передачі даних від вузла відправника до вузла одержувача крізь канали різних мереж. Цю задачу на мережевому рівні вирішують вузли-маршрутизатори, які одночасно є вузлами двох або більшої кількості суміжних мереж.
6. Задача маршрутизації, яка полягає в пошуку найкращого маршруту доставки *IP*-пакетів, вирішується на мережевому рівні за допомогою протоколів маршрутизації. При цьому не гарантується успішність доставки, бо за різних причин пакети можуть бути загублені.
7. У версії *IPv*6 введено новий параметр мітка потоку, що призначений для прискорення маршрутизації. Цей параметр зберігається в пам’яті маршрутизаторів протягом шести секунд. Весь цей час пакети з однаковими мітками потоку відправляються на той самий інтерфейс, що знайдений у таблиці маршрутів для першого пакету потоку.
8. Аналіз темпів впровадження *IPv*6, показує, що на червень 2022 року (10 років з початку переведення магістральних мереж Інтернету на версію *IPv*6) лише 28% ресурсів Інтернету підтримують нову версію. Суттєвого покращення характеристик роботи мережі Інтернет від переходу на нову версію протоколу *IP* не було виявлено і процес впровадження *IPv*6 на даний період часу є затяжним.

9. Надійний зв’язок між вузлами забезпечує протокол транспортного рівня *TCP*, який утворює наскрізне і прозоре сполучення але при цьому можуть виникати затримки через повторну передачу втрачених пакетів.
10. Не для всіх сеансів зв’язку є необхідність у абсолютно надійній доставці пакетів. У разі телевізійної або радіо трансляції знищення деяких пакетів може несуттєво впливати на якість передавання. Це у значній мірі стосується також Інтернет телефонії. У таких випадках використання протоколу *TCP* може призвести до негативних наслідків, бо виправлення помилок викличе затримку, що значно погіршить якість сприйняття інформації у реальному часі.
11. Крім протоколу *TCP* на транспортному рівні важливе місце займає протокол *UDP*. Цей протокол не гарантує надійність доставки даних, але забезпечує мінімальну затримку і надає можливість передавання широкомовних та термінових повідомлень.
12. З метою покращання якості зв’язку в умовах реального часу і перш за все для Інтернет телефонії на транспортному рівні створено нові протоколи, серед яких найбільш відомими є *SCTP* (*Stream Control Transmission Protocol* – протокол передачі з управлінням потоком) та *DCCP* (*Datagram Congestion Control Protocol* – протокол управління заторами пакетів).

Запитання та завдання для самоперевірки

1. Надайте пояснення необхідності створення протоколів мережевого та транспортного рівнів.
2. Які переваги протоколу *IPv*6 над *IPv*4?
3. Поясніть яким чином забезпечується можливість співіснування в мережі Інтернет версій протоколу *IPv*4 та *IPv*6.
4. Надайте приклади повного і скороченого варіантів запису адрес *IPv*6.
5. Надайте перелік адрес особливого призначення для протоколів *IPv*4 та *IPv*6.
6. Чим відрізняється процедура маршрутизації пакетів *IPv*6 над *IPv*4?
7. Яку роль відіграє протокол транспортного рівня *TCP*?
8. Чи для всіх сеансів зв’язку є необхідність у абсолютно надійній доставці пакетів?
9. Які особливості протоколу транспортного рівня *UDP*?
10. З якою метою створено протоколи транспортного рівня *SCTP* та *DCCP*?

Р О З Д І Л 4

АДРЕСАЦІЯ РЕСУРСІВ МЕРЕЖІ ІНТЕРНЕТ

4.1. Символьні адреси прикладного рівня

Мережа Інтернет забезпечує доступ до неосяжної кількості ресурсів. Кожному з цих ресурсу потрібна унікальна адреса. Крім того, адреси повинні бути такими, щоб користувачі мали змогу ними оперувати без особливих ускладнень. Цифрових адрес, які ми розглянули у попередніх розділах, недостатньо для розв'язання задачі адресації усіх ресурсів мережі Інтернет. Цифрові адреси необхідні для забезпечення обміну даними між фізичними та логічними об'єктами мережі, але вони не зручні для користувачів, яким легше оперувати символьними адресами. Ще на початку створення комп'ютерних мереж усім вузлам крім цифрової адреси надавали альтернативну символьну адресу (чи ім'я). Для перетворення символьних адрес у цифрові (і навпаки) створили файл *HOSTS.TXT*, у якому зберігалась таблиця відповідності символьних адрес цифровим. Зараз для цього перетворення використовується спеціалізоване програмне забезпечення системи доменних імен *DNS* (*Domain Name System*). Але і у сучасних операційних системах існує файл *hosts*, який функціонально відповідає файлу *HOSTS.TXT*. Найчастіше у файлі *hosts* є лише один рядок:

127.0.0.1 *localhost*

Місце файлу *hosts* для різних операційних систем можна дізнатись за посиланням: *https://en.wikipedia.org/wiki/Hosts_(file)*. До файлу *hosts* є доступ для модифікації, а оскільки йому надано вищий пріоритет у порівнянні із системою *DNS*, то цим файлом можна скористатись для направлення запитів користувача на *IP* адресу будь-якого сервера. Для зловмисників це надає змогу направляти запити на свій сервер із підробленим ресурсом, що схожий на справжній, але його задача може бути у розкритті паролів або наданні фальшивих даних. Тому корисно періодично перевіряти зміст файлу *hosts* на своїх комп'ютерах.

Розглянемо структуру символьних адрес ресурсів Інтернету, які мають назву *URI* (*Uniform Resource Identifier* — Уніфікований ідентифікатор ресурсу), на прикладі посилання, за яким надано роз'яснення цієї структури адрес: https://en.wikipedia.org/wiki/Uniform_Resource_Identifier#Syntax.

Елементи даної символьної адреси пронумеровані та описані на рис. 4.1.

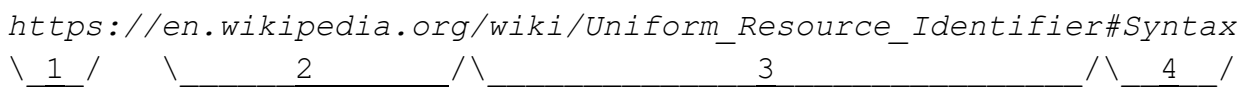


Рис. 4.1. Елементи символьної адреси ресурсу (1 – назва протоколу прикладного рівня, 2 – адреса чи ім'я вузла, 3 – ім'я файлу разом із директорією, 4 – назва розділу у файлі)

Важливою властивістю символьних адрес є придатність перетворення їх у цифрові (*IP* адреси та номери портів), які потрібні для заголовків пакетів мережевого та транспортного рівнів. Символьну адресу, що позначена номером 2 (див. рис. 4.1), можна перетворити у *IP* адресу за допомогою системи *DNS*, а порт для протоколу *https* по замовчанню буде мати закріплений номер 443 (див. табл. 1.15). Таким чином, маючи символьну адресу ресурсу у форматі *URI*, можна отримати усі необхідні дані для формування і передавання пакетів на мережевому та транспортному рівнях.

У форматі *URI* передбачено більше можливостей, ніж ті, що показані на даному прикладі. У разі коли володар ресурсу має бажання замість закріпленого номера порту надати інший номер, про що повинні знати тільки його визначені користувачі, то після адреси вузла слід поставити символ ":", а одразу після нього номер порту. Ще є можливість після ім'я файлу надати параметри запиту так, як це робиться у широко відомому перекладачу: *https://translate.google.com.ua/?sl=en&tl=uk&text=train&op=translate*, де після ім'я файлу стоїть символ "?" з низкою параметрів. Легко зрозуміти, що *sl=en* означає англійську мову джерела, *tl=uk* означає українську мову перекладу, *text=train* означає текст, який треба перекласти, а *op=translate* означає опцію перекладу. Можливості формату *URI* перевищують те, що потрібно для символьних адрес у мережі Інтернет, тому слід зауважити, що ми розглянули тільки деякі з усіх можливостей *URI*.

4.2. Система доменних імен *DNS*

До складу сучасної системи *DNS* відносять три основні компоненти.

- Розподілена база доменних імен (*DNS database*).
- Сервери імен (name server).
- Клієнтські програми визначення *IP* адрес (*name re-solver*).

Простір доменних імен нагадує деревоподібну файлову структуру, що показано на рис. 4.2.

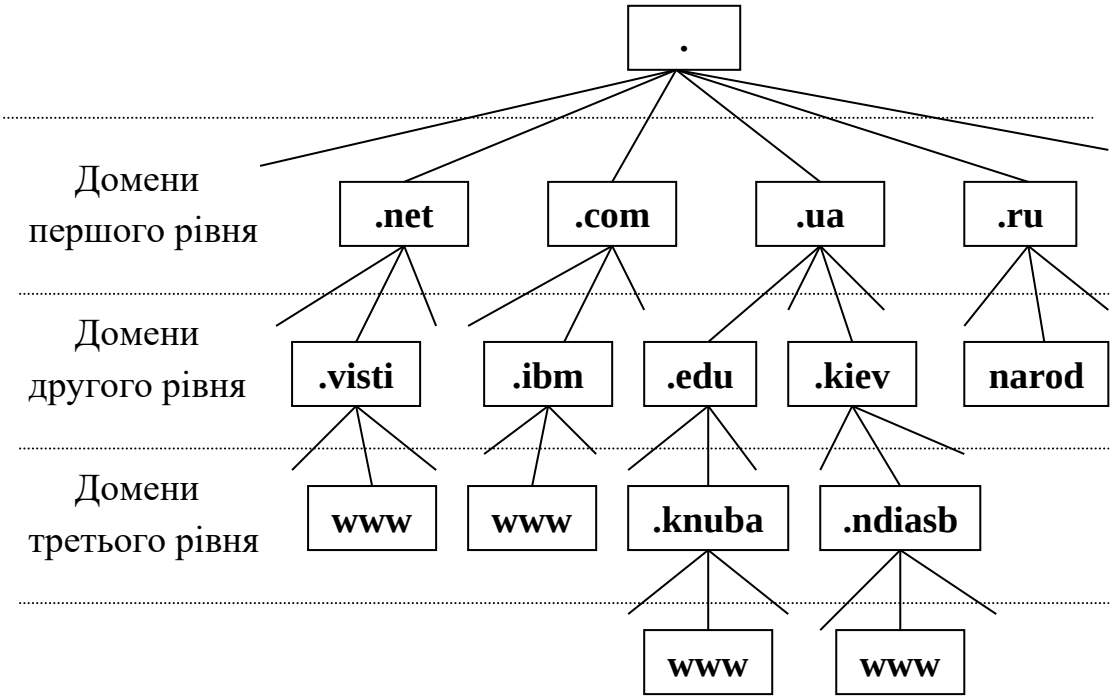


Рис. 4.2. Фрагмент структури простору доменних імен

Корінь цього дерева позначений символом “.” (крапка). Цей символ повинен стояти у кінці кожного доменного імені, але у форматі інтерфейсу користувача його не ставлять. У записах, що знаходяться в базі даних, відсутність цієї крапки є грубою помилкою.

Зміст бази даних *DNS* коригується вручну у текстових файлах, після чого за допомогою програми з цих файлів формуються рядки бази даних, що мають назву resources records (записи про ресурси). Ця база має такі шість полів:

NAME – ім’я ресурсу (255 байт).

TYPE – тип ресурсу.

CLASS – клас ресурсу.

TTL – час зберігання запису про ресурс у пам’яті користувачів.

RDLENGTH – довжина поля даних (число до 65535).

RDATA – дані (до 65535 байт).

Зразок текстового файлу для коригування записів у базі даних *DNS* зображено на рис. 4.3.

```
$TTL   86400
dim.kiev.ua. IN  SOA   ns2.ndiasb.kiev.ua. adm.ndiasb.kiev.ua. (
                        200304121;serial number
                        28800   ;refresh period
                        1800    ;retry refresh this often
                        604800  ;expiration period
                        86400   ;minimum TTL
                        )

             IN NS    ns2.ndiasb.kiev.ua.
             IN NS    ns2.elvisti.kiev.ua.

             IN MX 5  smtp.ndiasb.kiev.ua.
             IN MX 10   smtp2.visti.net.
             IN A     195.64.255.162

www          IN A     195.64.255.162
ftp          IN A     195.64.255.162
```

Рис. 4.3. Текстовий файл, що відповідає одній з зон бази даних *DNS*

У першому рядку цього файлу задано значення *TTL* у секундах.

У другому рядку знаходиться початок запису типу *SOA* (*Start of Authority*). З цього запису починається зона домену, ім'я якого стоїть на початку рядка (у полі *NAME*).

Параметр *IN* означає клас ресурсу (він відповідає полю *CLASS*). В нашому прикладі використовується тільки один клас *IN* (*Internet*).

Запис типу *SOA* у полі *RDATA* має сім параметрів, що відділяються один від одного пропусками. Перший параметр означає ім'я головного (*primary*) сервера *DNS*, на якому знаходиться ця зона. Другий – адреса електронної пошти особи, що відповідає за *DNS*-сервер. Для відправлення листів до цієї особи слід заміняти першу крапку у адресі на символ @. У кінці рядка стоїть ліва дужка, що свідчить про наявність продовження запису у наступних рядках до появи правої дужки. Наступні 5 числових параметрів можна було б написати у одному рядку без дужок. Вони записані у окремих рядках тільки для зручності читання. Значення числових параметрів ми розглянемо трохи нижче, а зараз звернемо увагу на крапки з комою. Вони означають кінець рядка. Кінцеві символи, включаючи крапку з комою, не заносяться у базу даних *DNS*.

Далі у рядках наведено записи наступних трьох типів.

NS – ім'я *DNS* серверу. Кількість таких записів дорівнює кількості серверів (головного та допоміжних), у яких розміщено цю зону DNS.

MX – ім'я сервера, на який слід відправляти електронну пошту. У цих записах числа 5 та 10 означають пріоритети. Числа вибирають які завгодно, але враховують, що меншому числу відповідає вищий пріоритет.

A – *IPv*4 адреса вузла, де знаходиться ресурс, ім'я якого вказано у полі *NAME*.

В усіх цих записах пропуски на початку рядку означають, що поле *NAME* буде скопійоване із запису *SOA*. Текст на початку рядка, що не закінчується крапкою, буде доповнено крапкою та копією поля *NAME* з запису *SOA*. Для останніх двох рядків поля *NAME* будуть виглядати, як *www.dim.kiev.ua*. та *ftp.dim.kiev.ua*. відповідно.

Для адрес протоколу *IPv*6 додано спеціальний тип записів:

AAAA – *IPv*6 адреса вузла, де знаходиться ресурс, ім'я якого вказано у полі *NAME*.

Сервери *DNS* відносно джерела інформації бувають:

- головними або первинними (*Primary Name Server*), у яких базу даних заповнюють та коригують вручну;
- допоміжними або вторинними (*Secondary Name Server*), у яких база даних регулярно копіюється з головного сервера;
- кешуючі (*Cache only Server*), що зберігають кешовану інформацію.

Головний та допоміжний сервери *DNS* повинні розміщуватись у різних мережах. Необхідно, щоб існував хоч один допоміжний сервер. Сервер може бути одночасно головним для одних зон та допоміжним для інших.

Функціонування системи *DNS* у разі сучасної версії програмного забезпечення *BIND v*.4.9 має такий вигляд.

Допоміжні сервери поновлюють свої дані від головного сервера згідно числовим параметрам запису *SOA* (див. рис. 4.3).

Перший числовий параметр, що має назву *serial number*, являє собою довільне число, яке слід змінювати під час коригування зони *DNS*.

Другий числовий параметр являє собою період запитів (у секундах) від допоміжного сервера до головного. Копіювання даних відбувається тільки у тому разі, коли виявляється заміна параметру *serial number*.

Третій числовий параметр являє собою період повторень запитів у разі невдалої спроби встановлення зв'язку (у секундах).

Четвертий числовий параметр обмежує період невдалих спроб до моменту їх остаточного припинення.

Останній числовий параметр обмежує знизу значення *TTL*.

Розглянемо процедуру визначення *IP* адреси за допомогою *DNS*.

Клієнтська програма робить запит до *DNS* сервера, що обслуговує мережу клієнта. Якщо у пам'яті сервера є відповідь на запит клієнта, він одразу відповідає. Інакше, сервер починає процедуру опитування інших серверів, починаючи з корінного. Адреси корінних серверів завжди відомі, бо вони надаються у програмному забезпеченні усіх серверів *DNS*.

За алгоритмом роботи сервери *DNS* можуть бути рекурсивними або ітеративними (не рекурсивними). Рекурсивними називають такі сервери, які можуть формувати запити до інших серверів з метою здобуття остаточної відповіді про *IP* адресу ресурсу. Ітеративні сервери не формують запити до інших серверів, а дають не повну відповідь у вигляді *IP* адреси наступного сервера, до якого слід звертатись за відповіддю.

Корінні сервери завжди ітеративні. Вони надають відповідь у вигляді списку *IP* адрес *DNS* серверів, які обслуговують домен першого рівня. Наприклад, якщо у запиті задано ім’я *www.dim.kiev.ua*, то у відповіді будуть адреси серверів домену *ua*, до яких слід звертатись із цим самим запитом.

Після одержання такої відповіді перший сервер, який є рекурсивним, формує черговий запит за *IP* адресами, що отримані у списку.

Так буде продовжуватись доти, поки не прийде остаточна відповідь у вигляді *IP* адреси або відмови. Цю відповідь сервер відправляє клієнту, а той використовує її під час формування заголовків *IP* пакетів (їх структуру показано у табл. 1.12).

З моменту свого виникнення в 1983 році і до недавнього часу *DNS* в основному відповідала на запити за протоколом *UDP* із номером порту 53. Такі запити складаються з відкритого тексту, надісланого в *UDP* пакеті від клієнта, та відповіді у вигляді відкритого тексту, надісланого також в *UDP* пакеті від сервера. Використання *DNS* у такому вигляді обмежується тим, що через відсутність шифрування на транспортному рівні можливі спроби втручання зловмисників у процес отримання адрес. У 1989 році визначено додатковий транспорт для *DNS* з використанням протоколу *TCP* (*RFC* 1123), що забезпечує надійну доставку та повторне використання довготривалих з’єднань між клієнтами та серверами. У 2016 році з’явився стандарт *IETF* для зашифрованого *DNS*, який використовує захист транспортного рівня для з’єднання із серверами *DNS* через *TCP*-порт 853. *RFC* 7858 визначає, що може підтримуватися шифрування та автентифація, але це не є обов’язковим. Протокол *DNSCrypt*, який був розроблений у 2011 році поза рамками стандартів *IETF*, запровадив шифрування *DNS* на нижній стороні рекурсивних засобів, де клієнти шифрують дані за допомогою відкритих ключів серверів, які публікуються в *DNS* (замість

того, щоб покладатися на центри сертифікації), які, у свою чергу, можуть бути захищені цифровими підписами. У 2019 році *DNSCrypt* було додатково розширено для підтримки «анонімного» режиму, у якому вхідний вузол отримує запит, який був зашифрований відкритим ключем іншого сервера. При цьому створюється конфіденційність пар користувач/запит, оскільки вхідний вузол не знає вмісту запиту, тоді як вихідні вузли не знають ідентичність клієнта. *DNSCrypt* був вперше реалізований у виробництві *OpenDNS* у грудні 2011 року. Існує кілька реалізацій безкоштовного програмного забезпечення з відкритим кодом, що доступний для різних операційних систем, включаючи *Unix, Apple iOS, Linux, Android* і *MS Windows*.

4.3. Адресація до ресурсів мережі з вузлів із внутрішніми адресами

Переважна більшість користувачів Інтернету мають лише внутрішні *IP* адреси і не відчувають при цьому труднощів з доступом до ресурсів мережі. Внутрішні *IP* адреси, які називають "сірими" або приватними чи локальними, на мові оригіналу мають назву *private address.* Перевагою цих адрес є те, що вони безкоштовні і для них не існує поняття дефіциту у порівнянні із зовнішніми ("білими", глобальними, реальними, публічними або *public address*). Крім того, до хостів із внутрішніми адресами зазвичай заборонено зовнішній доступ, що забезпечує їх захист від атак зловмисників. При цьому завдяки технології *NAT* (*Network address translation*) забезпечується доступ до зовнішніх ресурсів з хостів із внутрішніми адресами. Технологія *NAT* використовується у декількох варіантах, з яких найбільш розповсюджений називають *NAPT* або *PAT* (*Port Address Translation*). Цей варіант дозволяє через одну "білу" *IP* адресу надати доступ до зовнішніх ресурсів десяткам тисяч користувачів із "сірими" адресами. Для цього маршрутизатор, у якому реалізовано технологію *PAT* пересилає пакети із внутрішньої мережі до зовнішньої, із заміною *IP* адреси відправника (внутрішньої) у заголовку *IP* пакету на якусь зі своїх публічних адрес. Крім того, порт відправника у заголовку *TCP* (або іншого протоколу транспортного рівня) маршрутизатор замінює на один зі своїх вільних портів. Після цих замін пакет відправляється у магістральну мережу. Зроблені заміни заносять у базу даних, структура якої представлена у табл. 4.1.

Таблиця 4.1

Структура бази даних технології *PAT*

Дані відправника пакету на вході маршрутизатора		Дані відправника цього пакету після заміни на орендовані		Час закінчення оренди
IP адреса	порт	*IP* адреса	порт	
192.168.1.102	49162	91.198.50.120	61148	
192.168.1.105	50672	91.198.50.120	61149	
192.168.1.108	60234	91.198.50.120	61151	

Період оренди встановлюють з урахуванням максимально можливого часу очікування відповіді від сервера, що зазвичай дорівнює декілька хвилин. У разі повторного запиту з хоста внутрішньої мережі (з тих самих *IP* адреси та порту) час оренди продовжують.

Крім налаштування технології *NAT* на оренду однієї реальної *IP* адреси для усіх запитів із внутрішньої мережі є можливість надавати в оренду декілька реальних адрес по черзі. Такий метод називають *Dynamic NAT* (динамічний *NAT*). Він дозволяє зняти обмеження доступу із внутрішньої мережі до ресурсів, на яких дозволено з однієї *IP* адреси користуватись у кожний момент часу лише одному користувачеві.

З метою захисту хостів внутрішньої мережі від небажаних проникнень через орендовану *IP* адресу використовують доповнену структуру бази даних, куди заносять ще *IP* адресу ресурсу, до якого відправляється запит. При цьому маршрутизатор не дозволяє проходження пакетів з інших *IP* адрес через надані в оренду *IP* адресу і порт. З метою посилення захисту хостів внутрішньої мережі від зайвих проникнень у базу даних, крім *IP* адреси одержувача пакету, можуть заносити ще й номер його порту. Тоді відправник зможе отримувати відповіді на свій запит тільки з тієї *IP* адреси і того порту, куди було відправлено запит.

У разі коли користувач хоста у внутрішній мережі бажає розмістити в себе загально доступний ресурс, то за допомогою технології *NAT* йому може бути надано реальну *IP* адресу у довгострокову оренду. При цьому пакети, що надійдуть на цю *IP* адресу, маршрутизатор пересилатиме на внутрішню адресу користувача. Цей користувач буде залишатись зі своїми правами у внутрішній мережі, а зовнішній доступ до нього буде таким, як до хоста із реальною адресою.

Достатньо велика кількість *IP* адрес, що зарезервовані для внутрішніх мереж, дозволяє у внутрішніх мережах створювати додаткові внутрішні мережі без будь-яких суттєвих обмежень щодо використання технології *NAT*.

Таким чином, завдяки розвитку технології *NAT* вдалося подолати такий важливий недолік протоколу *IPv*4, як обмеження кількості адрес, а також захистити вузли внутрішніх мереж від небажаних проникнень з магістральної мережі.

В и с н о в к и

1. Цифрові адреси, які необхідні для обміну даними між вузлами мережі, не задовольняють вимогам щодо адресації усіх ресурсів та послуг у мережі Інтернет. Користувачам мережі зручніше оперувати символьними адресами ніж цифровими. Завдяки впровадженню символьних адрес, які побудовані за форматом *URI* (*Uniform Resource Identifier*), розв'язано задачу адресації зростаючої кількості ресурсів та послуг мережі Інтернет.
2. Важливою властивістю символьних адрес за форматом *URI* є їх можливість перетворення у цифрові (*IP* адреси та номери портів), які потрібні для формування пакетів мережевого та транспортного рівнів. Це перетворення зараз виконується за допомогою системи доменних імен *DNS* (*Domain Name System*) та стандарту щодо призначення портів.
3. На початку створення мережі Інтернет (тоді її називали *ARPANET*) усім вузлам крім *IP* адреси надавали альтернативну символьну адресу. Для перетворення символьних адрес у цифрові використовували таблицю у файлі *HOSTS.TXT*. У сучасних системах є файл *hosts*, який функціонально відповідає файлу *HOSTS.TXT* та має пріоритет вищий за *DNS*.
4. Оскільки файлу *hosts* надано найвищий пріоритет, то цим можуть скористатись зловмисники для направлення запитів користувача на *IP* адреси шкідливих ресурсів. Тому корисно перевіряти зміст файлу *hosts* на своїх комп'ютерах.
5. Простір доменних імен системи *DNS* нагадує деревоподібну файлову структуру. Головна задача цієї системи – знаходження *IP* адрес серверів, на яких розміщено ресурси, що мають задане доменне ім’я. До складу системи *DNS* відносять три основні компоненти.
   - Розподілена база доменних імен (*DNS database*).
   - Сервери імен (name server).
   - Клієнтські програми визначення *IP* адрес (*name re-solver*).
6. Сервери *DNS* відносно джерела інформації бувають:
   - головними або первинними (*Primary Name Server*), у яких базу даних заповнюють та коригують вручну;

- допоміжними або вторинними (*Secondary Name Server*), у яких база даних регулярно копіюється з головного сервера;

кешуючі (*Cache only Server*), що зберігають кешовану інформацію.

7. Переважна більшість користувачів Інтернету мають лише внутрішні *IP* адреси і не відчувають при цьому труднощів з доступом до ресурсів мережі. Внутрішні *IP* адреси, які називають "сірими" або приватними чи локальними, на мові оригіналу мають назву *private address*. Перевагою цих адрес є те, що вони безкоштовні і для них не існує поняття дефіциту у порівнянні із зовнішніми ("білими", глобальними, реальними, публічними або *public address*).

8. До хостів із внутрішніми адресами зазвичай заборонено зовнішній доступ, що забезпечує їх захист від атак зловмисників. При цьому завдяки технології *NAT* (*Network address translation*) забезпечується доступ до зовнішніх ресурсів з хостів із внутрішніми адресами.

9. Технологія *NAT* використовується у декількох варіантах, з яких найбільш розповсюджений називають *NAPT* або *PAT* (*Port Address Translation*). Цей варіант дозволяє через одну "білу" *IP* адресу надати доступ до зовнішніх ресурсів десяткам тисяч користувачів із "сірими" адресами.

10. Крім налаштування технології *NAT* на оренду однієї реальної *IP* адреси для усіх запитів із внутрішньої мережі є можливість надавати в оренду декілька реальних адрес по черзі. Такий метод називають *Dynamic NAT* (динамічний *NAT*). Він дозволяє зняти обмеження доступу із внутрішньої мережі до ресурсів, на яких дозволено з однієї *IP* адреси користуватись у кожний момент часу лише одному користувачеві.

11. У разі коли користувач хоста у внутрішній мережі бажає розмістити в себе загально доступний ресурс, то за допомогою технології *NAT* йому може бути надано реальну *IP* адресу у довгострокову оренду. При цьому пакети, що надійдуть на цю *IP* адресу, маршрутизатор пересилатиме на внутрішню адресу користувача. Цей користувач буде залишатись зі своїми правами у внутрішній мережі, а зовнішній доступ до нього буде таким, як до хоста із реальною адресою.

12. Завдяки розвитку технології *NAT* вдалося подолати такий важливий недолік протоколу *IPv*4, як обмеження кількості адрес, а також захистити вузли внутрішніх мереж від небажаних проникнень з магістральної мережі .

Запитання та завдання для самоперевірки

1. Надайте пояснення щодо необхідності впровадження символьних адрес і їх перетворення у цифрові адреси.
2. Які переваги та обмеження надає використання внутрішніх *IP* адрес у порівнянні із публічними *IP* адресами?
3. Поясніть яким чином забезпечується доступ користувача з внутрішньою *IP* адресою до зовнішніх ресурсів Інтернету?
4. Надайте приклад формування символьних адрес за стандартом *URI* та поясніть яким чином відбувається їх перетворення у цифрові адреси.
5. Яку роль відіграє технологія *NAT* із врахування різноманітних варіантів її використання.
6. Надайте перелік блоків *IP* адрес, що призначені для використання у внутрішніх мережах.
7. Яку роль відіграє файл *hosts* у сучасних системах та чи варто перевіряти його зміст?
8. З якою метою було створено систему *DNS*, які її складові та принципи функціонування?
9. Чи впливає розвиток технології *NAT* на уповільнення впровадження протоколу *IPv*6?

Список літератури

1. *Вишняков В.М.* Основи побудови комп'ютерних мереж: Навчальний посібник – К.: КНУБА, 2013. – 128 с.
2. *Жураковський Б.Ю., Зенів І.О.* Комп'ютерні мережі Частина 1: Навчальний посібник [Електронний ресурс] – Київ : КПІ ім. Ігоря Сікорського, 2020. – 336 с.
3. *Самофалов К.Г., Романкевич А.М., Валуйский В.Н., Каневский Ю.С., Пиневич М.М.* Прикладная теория цифровых автоматов – К.: Вища школа, 1987. – 375 с.
4. *Котельников В.А.* Теория потенциальной помехоустойчивости. – М.: Госэнергоиздат, 1956. – 152 с.
5. Частотно-фазовый манипулятор: авторское свидетельство СССР №836816 М.кл. H 04 L 27/12 / Вышняков В.М., Гирнык А.В., Захарченко В.И., Кириевский Л.А.; заявл. 31.07.1979; опубл. 07.06.1981, Бюл. №21. 2 с.
6. *Возенкрафт Д., Джекобс И.* Теоретические основы техники связи. – М.: Мир, 1969. – 640 с.
7. *Troubleshooting Ethernet – Cisco. Retrieved May* 18, 2021. *https://www.cisco.com/en/US/docs/internetworking/troubleshooting/guide/tr1904.html*
8. *High-Speed Transmission Update*: 200*G*/400*G – Connectorsupplier. Retrieved* 2022-08-20. *https://connectorsupplier.com/high-speed-transmission-update-200g400g/*
9. *Ethernet Roadmap* 2022 *– Ethernet Alliance*. 2022. *Retrieved* 2022-08-20. *https://ethernetalliance.org/technology/ethernet-roadmap/*
10. *RFC* 2460. *Internet Protocol, Version* 6 (*IPv*6) *Specification.:IETF, December* 1998.
11. *RFC* 5156. *Special-Use IPv*6 *Addresses.: IETF, April* 2008.
12. *RFC* 6434. *IPv*6 *Node Requirements.: IETF, December* 2011.
13. *Internet Protocol Version* 4 (*IPv*4) *Parameters – IANA Last Updated 2018-05-03. Retrieved* 2022-08-20. *https://www.iana.org/assignments/ip-parameters/ip-parameters.xhtml/*
14. *RFC* 3068. *An Anycast Prefix for 6to4 Relay Routers.:IETF, June* 2001.
15. *RFC* 5969. *IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) - Protocol Specification.: IETF, August* 2010.
16. *RFC* 4380. *Teredo: Tunneling IPv*6 *over UDP through Network Address Translations (NATs).: IETF, February* 2006.
17. *RFC* 6081. *Teredo Extentions.: IETF, January* 2011.
18. *Balchunas A. Routing Information Protocol* (*RIP v1.03*) *– Routeralley*. 2012. *Retrieved* 2022-08-20. *http://www.routeralley.com/guides.html/*

19. *IP Routing: OSPF Configuration Guide, Cisco IOS Release* 15*SY – Cisco. Retrieved November* 14, 2021. *https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_ospf/configuration/15-sy/iro-15-sy-book/iro-sham-link.html/*
20. *Enhanced Interior Gateway Routing Protocol* (*EIGRP*) *– Cisco Systems* (2013) *https://www.cisco.com/c/en/us/products/ios-nx-os-software/enhanced-interior-gateway-routing-protocol-eigrp/index.html/*
21. *Border Gateway Protocol* (*BGP*) *– Cisco. Retrieved* 2022-08-20. *https://www.cisco.com/c/en/us/products/ios-nx-os-software/border-gateway-protocol-bgp/index.html/*
22. *Ford M. An Eye On The Numbers: IPv6 Deployment – Internet Society Pulse. Retrieved* 2022-08-20. *https://pulse.internetsociety.org/blog/an-eye-on-the-numbers-ipv6-deployment*

Додаток 1

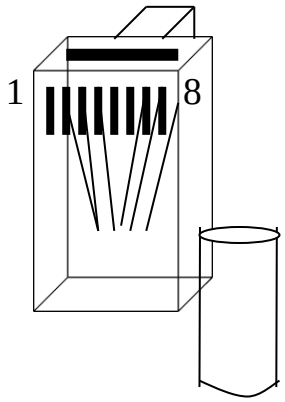
Організації, що розробляють стандарти комп'ютерних мереж

Назва організації	Перелік розробок	Приклад позначення стандарту
International Organization for Standardization, ISO, або *International Standard Organization* (Міжнародна організація зі стандартизації)	Модель взаємодії відкритих систем (*Open System Interconnection, OSI*), що стала концептуальною основою стандартизації у галузі КМ	7498 *ISO*
International Telecommunications Union, ITU (Міжнародний телекомунікаційний союз)	Видання серій рекоменда-цій-стандартів. Серія *V* – передавання даних по телефонних каналах, серія *X* – мережі передавання даних	*V*.90 *X* .500
Institute of Electrical and Electronics Engineers, IEEE (Інститут інженерів електротехніків та електро-ніків)	Стандарти та вимоги до локальних КМ. Стандартизація найбільш розповсюджених технологій *Ethernet* та *Token Ring*	*IEEE* 802.5
European Computer Manufacturers Association, ECMA (Європейська асоціація виробників комп'ютерів)	Стандарти передавання графічних зображень та текстів зі збереженням оригінально-го формату	*ECMA*-101
Electronic Industries Association, EIA (Асоціація електронної промисловості)	Стандарти інтерфейсів, кабелів та роз'єднувачів	*EIA* *RS*-232*C*
Internet Engineering Task Force, IETF (Група, що вирішує технічні проблеми мережі Інтернет)	Під керівництвом Internet Society (ISOC) – компанії, що займається розвитком мережі Інтернет, група роз-глядає пропозиції та надає статус стандартів технічним рішенням мережі Інтернет.	*RFC* 1700

Розміщення кінців скручених пар у роз'єднувачах типу 8*P*8*C* (*RJ*-45)

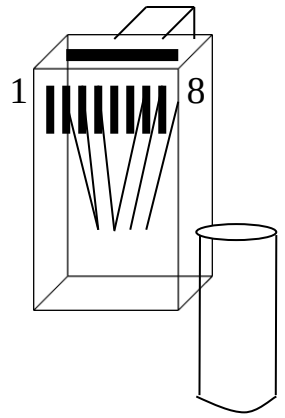
Стандарт *EAI/TIA*-586*A*

Номер	Колір проводу
1	біло-зелений
2	зелений
3	біло-оранжевий
4	синій
5	біло-синій
6	оранжевий
7	біло-коричневий
8	коричневий



Стандарт *EAI/TIA*-586*B*
та AT&T 258A

Номер	Колір проводу
1	біло-оранжевий
2	оранжевий
3	біло-зелений
4	синій
5	біло-синій
6	зелений
7	біло-коричневий
8	коричневий



Кабель для з'єднання двох комп'ютерів між собою без концентратора або комутатора повинен мати один роз'єднувач за стандартом EAI/TIA-586A, а другий - за стандартом EAI/TIA-586B. У інших випадках кінці розміщують за одним стандартом (рекомендовано EAI/TIA-586A).

Додаток 3

Формули для обчислення пропускної здатності каналів зв'язку

Для аналогових систем пропускну здатність обчислюють за допомогою формули Клода Шеннона $C = F \log_2(1+P_s/P_n)$, де C – пропускна здатність у бітах за секунду, F – ширина робочої смуги частот фізичного середовища передавання сигналів у Гц, P_s – середня потужність сигналу, P_n – середня потужність адитивної завади типу "білого" шуму [7].

Ця формула отримана в умовах дії адитивної завади, частотні складові якої розміщені рівномірно у межах робочої смуги частот. Така завада є у всіх фізичних середовищах, бо вона зумовлена хаотичним (тепловим) рухом молекул. Тривалість сигналу в доведенні цієї формули була покладена нескінченною, що не дозволяє досягти значення C у реальних системах зв'язку. За допомогою формули встановлюється чисто теоретичне обмеження можливостей того чи іншого каналу зв'язку.

У цифрових системах пропускну здатність можна знайти за формулою Найквиста $C = 2F \log_2 M$, де M – кількість варіантів значень інформаційного параметра сигналу, що відрізняють у системі зв'язку. Для бінарних систем, де відрізняють два варіанти 0 або 1, пропускна здатність за Найквистом дорівнює $2F$.

Схожий результат отримано В.О.Котельниковим, який довів, що у каналі, який має робочу смугу частот F, можна відрізнити не більше ніж $2F$ значень (відліків) аналогового сигналу. Для імпульсних систем це означає, що частота послідовності імпульсів у каналі зв'язку не може перевищити значення $2F$, бо інакше імпульси не можна буде відрізнити один від одного [4].

Ще один важливий результат випливає з теореми В.О.Котельникова. Це можливість повного відновлення форми аналогового сигналу, якщо маємо значення його відліків з частотою $2F$, при умові, що сигнал не мав гармонік з частотою більшою за F. Цей висновок покладено в основу усіх систем цифрового зв'язку (цифрові телефонія і телебачення), а також цифрових систем збереження та відтворення звуку і зображення.

Спектральний аналіз сигналів

Послідовність імпульсів з періодом T можна розкласти у ряд Фур'є
$u(t)=A_0+A_1\cdot\sin(\pi t/T+\varphi_1)+A_2\cdot\sin(2\pi t/T+\varphi_2)+\ldots+A_N\cdot\sin(N\pi t/T+\varphi_N)+\ldots$,
де N=1, 2, …, ∞. Це схематично зображено на рис.Д.1.

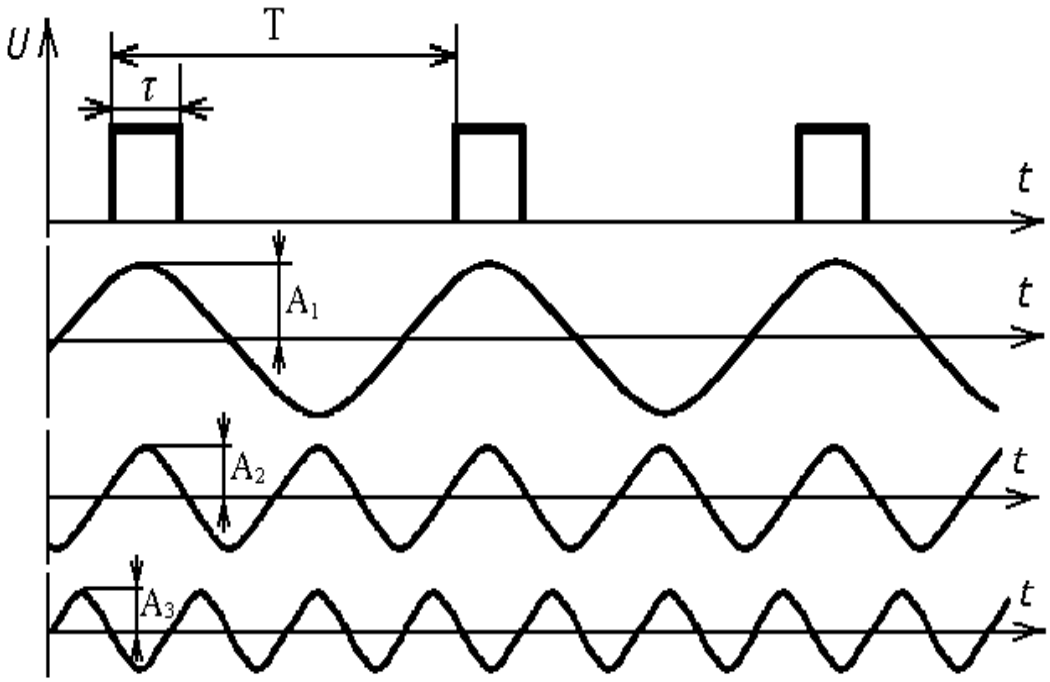


Рис.Д.1. Представлення послідовності імпульсів у вигляді суми синусоїд

Графік, на якому представлені амплітуди цих синусоїд залежно від частоти, називають дискретним спектром (рис. Д.2).

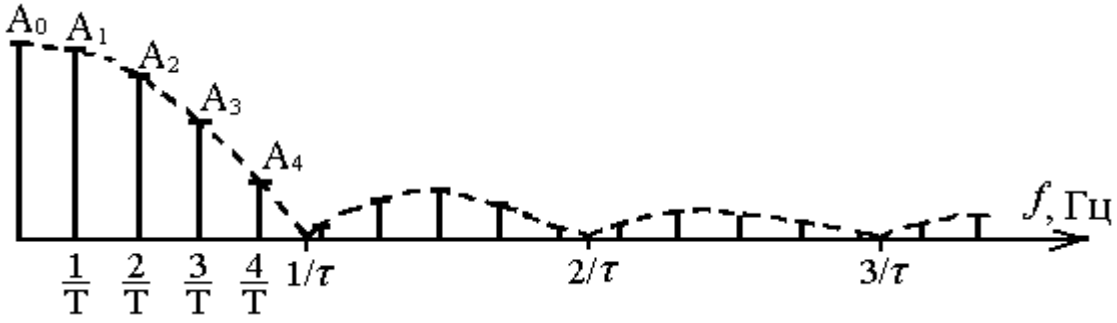


Рис.Д.2. Спектр послідовності імпульсів тривалістю τ з періодом T

Синусоїдальні складові сигналу називають гармоніками.

Спектр окремого імпульсу тривалістю τ можна отримати, поклавши T→ ∞. При цьому ряд Фур'є перетворюється у інтеграл Фур'є, а спектр замість дискретного стане безперервним (рис. Д.3).

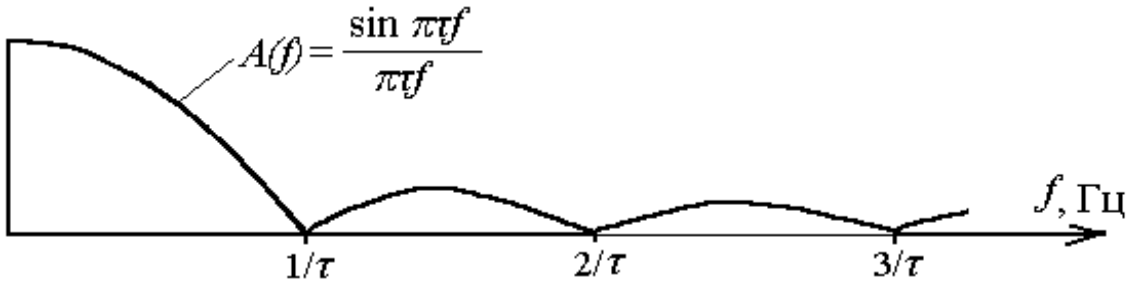


Рис. Д.3. Спектр окремого імпульсу тривалістю τ

Інтегруючи квадрат функції A(*f*) в інтервалах [0, 1/τ] та (1/τ, ∞), можна впевнитись, що більше ніж 90% енергії імпульсу зосереджено у смузі частот від 0 до 1/τ. Вважається, що поза цією смугою ослаблення спектра сигналів може бути будь-яким, бо це не чинить суттєвого впливу на процес передавання інформації. Ділянки спектра між нульовими значеннями називають пелюстками.

Якщо канал зв'язку забезпечує передавання гармонік у смузі частот *F*=1/τ, а імпульс тривалістю τ передає 1 біт інформації, то швидкість передавання становитиме 1 біт/с на кожен 1 Гц смуги частот. Це значення швидкості у 2 рази менше, ніж пропускна здатність за формулою Найквиста.

Метод досягнення максимальної швидкості передавання імпульсів у каналах з обмеженою смугою частот було запропоновано в роботі В.А. Котельникова [4]. В основу цього методу покладено узгодження форми імпульсних сигналів з частотною характеристикою каналу зв'язку. Для знаходження таких сигналів запропоновано скористатись реакцією каналу на імпульс, що має форму дельта-функції (голчатої функції). Уявіть собі імпульс з енергією (площею), що дорівнює одиниці у нескінченно малому проміжку часу. Зрозуміло, що амплітуда такого імпульсу буде нескінченно великою, а спектр – рівномірним у нескінченному діапазоні частот.

Якщо канал має обмежену смугу частот з прямокутною формою АЧХ, то сигнал, що є відгуком каналу на дельта-функцію, можна знайти за допомогою математичних обчислень. При цьому форма спектра сигналу буде співпадати з формою АЧХ. Результат обчислень для даного випадку зображений на рис. Д.4.

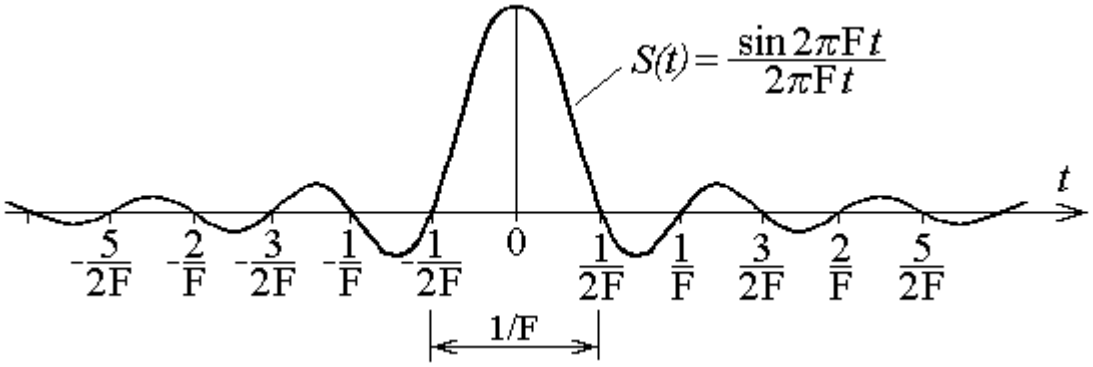


Рис. Д.4. Форма ідеального сигналу для каналу зі смугою частот F

Імпульси такої форми можна передавати через проміжки часу 1/2F, тобто з частотою 2F, що надає змогу досягти пропускної здатності за формулою Найквиста (рис. Д.5).

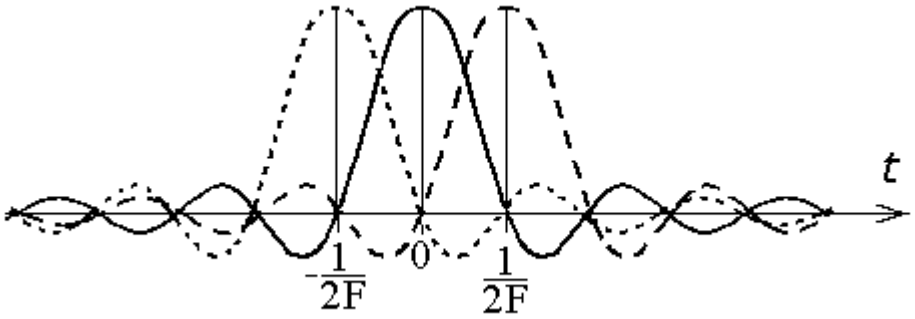


Рис. Д.5. Ілюстрація можливості передавання імпульсів з частотою 2F

Хоч у каналі зв’язку буде присутня сума переданих імпульсів, у ті моменти, коли вимірюється значення амплітуди чергового імпульсу, сума поточних значень усіх інших імпульсів дорівнює нулю.

Як бачимо, імпульси В.О. Котельникова мають нескінчену довжину, тобто займають у часі інтервал від – ∞ до + ∞. Це є наслідком обмеження спектра. Усі процеси, що мають початок або обмежені у часі, можуть мати тільки нескінчений спектр.

Один з основних висновків спектрального аналізу сигналів полягає в тому, що між тривалістю імпульсного сигналу та смугою частот, яку необхідно виділити у каналі зв’язку для передавання цього сигналу, існує зворотна пропорційність.

Додаток 5

Перелік скорочень

ANSI (*American National Standard Institute*) Американський національний інститут стандартів

ARP (*Address Resolution Protocol*) протокол перетворення мережевої адре- си у фізичну

ARPA (*Advanced Research Project Agency*) Агентство дослідження та роз- робки перспективних проектів

ASCII (*American Standard Code for Information Interchange*) американський стандартний код для обміну інформацією

ATM (*Asynchronous Transfer Mode*) режим асинхронного передавання

AUI (*Attachment Unit Interface*) інтерфейс для приєднання комп'ютера до приймача-передавача сигналів

BGP (*Border Gateway Protocol*) протокол граничного шлюзу

BIOS (*Basic Input/Output System*) базова система введення-виведення

CCITT (*Consultative Committee on International Telegraphy and Telephony*) Міжнародний консультативний комітет з телеграфії та телефонії

CIDR (*Classless Inter-Domain Routing*) безкласова міждоменна маршрути- зація

CRC (*Cyclic Redundancy Check*) циклічна контрольна сума

CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*) груповий доступ з контролем носія та виявленням колізій

DCCP (*Datagram Congestion Control Protocol* – протокол управління заторами пакетів)

DCE (*Data Circuit terminating Equipment*) апаратура передавання даних

DHCP (*Dynamic Host Configuration Protocol*) протокол динамічного надання мережних параметрів

DNS (*Domain Name System*) система доменних імен

DSL (*Digital Subscriber Line*) цифрова абонентська лінія

ECMA (*European Computers Manufacturers*) Європейська асоціація вироб-ників комп'ютерів

EGP (*Exterior Gateway Protocol*) протокол зовнішньої маршрутизації

EIA (*Electronic Industries Association*) Асоціація електронної промисловості

EoC (*Ethernet over Coaxial*) *Ethernet* крізь систему кабельного телебачення

FDDI (*Fiber Distributed Data Interface*) волоконно-оптичний інтерфейс для передачі даних

FTTB (*Fiber to the Building*) оптичне волокно до будинку

FTTC (*Fiber to the Curb*) оптичне волокно до мікрорайону

FTTH (*Fiber to the Home*) оптичне волокно до житла

FTTN (*Fiber to the Node*) оптичне волокно до вузла мережі

FTP (*File Transfer Protocol*) протокол передавання файлів

GPRS *(General Packet Radio Service)* загальний сервіс пакетного переда-вання у радіоканалах

GSM *(Global System for Mobile Communication)* глобальна система мо-більного зв'язку

HTTP *(Hypertext Transfer Protocol)* протокол передавання гіпертексту

ICMP *(Internet Control Message Protocol)* протокол діагностичних повідом-лень про стан мережі

IEC (*International Electrotechnical Committee*) Міжнародний комітет з елек-тротехніки

IEEE (*Institute of Electrical and Electronic Engineers*) Інститут інженерів зі електротехніки та електроніки у США

IEFT (*Internet Engineering Task Force*) Група, що вирішує технічні пробле-ми мережі Інтернет

IP (*Internet Protocol*) протокол зв'язку між мережами

ISDN (*Integrated Service Digital Network*) цифрова мережа з інтегруванням послуг

ISO (*International Organization for Standardization*) Міжнародна організація зі стандартизації

ISP (*Internet Service Provider*) посередник, який надає послуги доступу до мережі Інтернет

ITU (*International Telecommunication Union*) Міжнародний телекомуніка-ційний союз

MAC (*Media Access Control*) керування доступом до середовища

MLD (*Multicast Listener Discovery*) виявлення вузлів, що сприймають групові адреси

NetBEUI (*NetBIOS Extended User Interface*) розширений інтерфейс кори-стувача *NetBIOS*

NetBIOS (*Network BIOS*) мережева базова система введення-виведення даних

PDV (*Path Delay Value*) найбільша затримка обертання

POF (*Plastic Optical Fiber*) пластикове оптичне волокно

PON (*Passive Optic Network*) пасивна оптична мережа

RFC (*Request For Comments*) пропозиція для обговорення, що приймається *IETF* і після надання їй номера може набувати статусу стандарту

SCTP (*Stream Control Transmission Protocol* – протокол передачі з управлінням потоком)

SFP (*Small Form-factor Pluggable*) змінний малогабаритний модуль

TCP (*Transmission Control Protocol*) протокол управління передачею

UDP (*User Datagram Protocol*) протокол дейтаграм користувача

WDM (*Wave Division Multiplexing*) мультиплексування зі розподілом по довжині хвилі

WLAN (*Wireless Local Area Network*) бездротові локальні мережі

WMAN (*Wireless Metropolitan Area Network*) бездротові регіональні мережі

WPAN (*Wireless Personal Area Network*) бездротові персональні мережі

Навчальне видання

ВИШНЯКОВ Володимир Михайлович

ПРИНЦИПИ ПОБУДОВИ КОМП'ЮТЕРНИХ МЕРЕЖ

Навчальний посібник

Редагування та коректура *В.М. Вишняков*

Комп’ютерне верстання *М.М. Власенко*

Підписано до друку 26.12.2022. Формат 60x84$_{1/16}$.
Ум. друк. арк. 7,21. Обл.-вид. арк. 5,47.
Тираж 80 прим. Вид. № 16/І-17. Зам. № 15/1- 18

Видавець і виготовлювач
Київський національний університет будівництва і архітектури

Повітрофлотський проспект, 31, Київ, Україна, 03680

Свідоцтво про внесення до Державного реєстру суб’єкту
Видавничої справи ДК №808 від 13.02.2002