І. А. ЖУКОВ, В. І. ДРОВОВОЗОВ, Б. Г. МАСЛОВСЬКИЙ

Експлуатація комп'ютерних систем та мереж

Навчальний посібник

І. А. ЖУКОВ, В. І. ДРОВОВОЗОВ,
Б. Г. МАСЛОВСЬКИЙ

Експлуатація комп'ютерних систем та мереж

Навчальний посібник

Рекомендовано
Міністерством освіти і науки України
як навчальний посібник для студентів
вищих технічних навчальних закладів

Київ
Книжкове видавництво
Національного авіаційного університету
2007

УДК 004.7(076.5)
ББК 3973.202-082я7
Ж 86

Рецензенти:

О.М. Різник, д-р техн. наук
(Інститут проблем математичних машин та систем)

М.А. Виноградов, д-р техн. наук, проф.
(Національний авіаційний університет)

Гриф надано Міністерством освіти і науки України
(Лист № 14/182-426 від 21.02.06)

Видання друкується за рішенням
Вченої ради НАУ
Протокол № 4 від 19.04.06

Жуков І.А., Дрововозов В.І., Маслов[illegible]кий Б.Г.

Ж 86 Експлуатація комп'ютерних систем та мереж: Навч. посібник. — К.: НАУ, 2007. — 368 с.

ISBN 978-966-598-385-9

Навчальний посібник призначений для вивчення дисципліни «Експлуатація комп'ютерних систем та мереж» напряму «Комп'ютерна інженерія». Посібник підготовлений для роботи за кредитно-модульною системою.

Посібник складається з двох модулів, зміст яких тісно пов'язаний один з одним. Розглянуто основні поняття експлуатаційного обслуговування комп'ютерних систем та мереж, методи їх діагностування, методи адміністрування користувачів з використанням локальних і глобальних груп, підвищення експлуатаційної надійності комп'ютерних систем, технічні засоби експлуатаційного обслуговування і сучасні напрями розвитку технології експлуатаційного обслуговування. Наведено контрольні запитання, лабораторні роботи з прикладами виконання завдань.

Для студентів вищих технічних навчальних закладів.

УДК 004.7(076.5)
ББК 3973.202-082я7

ISBN 978-966-598-385-9

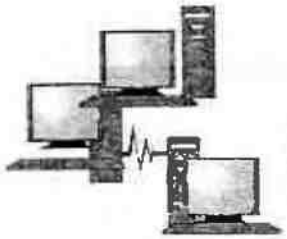
Вступ

Навчальний посібник призначено для вивчення дисципліни «Експлуатація комп'ютерних систем та мереж» напряму «Комп'ютерна інженерія». Дана дисципліна є основою сукупності знань та вмінь, що дозволяє спеціалістам запроваджувати експлуатаційне обслуговування комп'ютерних систем та мереж, їх програмне забезпечення.

Метою викладання дисципліни є розкриття сучасних методів експлуатації ЕОМ, комп'ютерних систем та мереж, дослідження методів підвищення надійності та підтримки експлуатаційного обслуговування.

Організація експлуатації комп'ютерних систем та мереж вимагає кваліфікованого інженерно-технічного персоналу, добре організованої служби матеріально-технічного постачання, значних експлуатаційних витрат. Теорія надійності дає змогу вивчати закономірності виникнення пошкоджень і відмов, процесів відновлення працездатності, методів підвищення надійності технічних і програмних засобів. Теорія експлуатації полягає у вивченні методів забезпечення необхідного рівня надійності й ефективності функціонування засобів у конкретних умовах їх використання.

Посібник підготовлено для роботи за кредитно-модульною системою і складається з двох модулів, зміст яких тісно пов'язаний один з одним. Розглянуто основні поняття експлуатаційного обслуговування технічних засобів комп'ютерних систем та мереж, методи їх діагностування, методи адміністрування користувачів із використанням локальних і глобальних груп, питання експлуатаційної надійності комп'ютерних систем, технічні засоби експлуатаційного обслуговування та сучасні напрями розвитку технології експлуатаційного обслуговування.



Модуль 1

Системи контролю функціонування комп'ютерних систем та мереж

1.1. Кодування інформації при передаванні її в каналах, де діє шум

1.1.1. Шумостійкість завдяки надлишковості

Вивчаючи основи теорії про суть сигналів, їх проходження через канали зв'язку, вплив шумів різної природи та походження, можна дійти висновку, що внаслідок цих перешкод сигнали зазнають значних спотворень. Сильні спотворення призводять до того, що збільшується частота неправильного «розпізнавання» приймачем за такими сигналами суті переданого (тобто деякі нулі сприймаються як одиниці і навпаки). Таким чином, сильні спотворення сигналів у дискретних каналах суттєво знижують шумності й якість пристроїв передавання інформації. Щоб подолати цю проблему, необхідно, по-перше, на основі теоретичних висновків (зокрема теорії потенційної шумостійкості Котельникова) вибирати оптимальні рішення побудови як системи передавання інформації в цілому, так і окремих її складових, обґрунтовано вибирати типи модуляції (в тому числі і відносних методів) з урахуванням як економічних, так і інших показників (шумостійкість, пропускна здатність і т. ін.); по-друге, різними методами захисту від появи в каналі або проникнення в нього шумів зменшувати їх інтенсивність, що пов'язано з суттєвими економічними витратами.

Але в ідеальному випадку, коли, припустімо, економічними показниками можна знехтувати, рівень спотворень і, як наслідок, імовірність помилкового сприйняття нулів та одиниць (елементарні помилки) була б такою, що про реальне використання систем передавання інформації не могло б бути й мови. Це тому, що вимога до достовірності прийнятих повідомлень має бути надто високою. Так, для систем передавання даних звичайного призначення

ймовірність прийняття неправильних знаків (символів, які кодуються вісьмома двійковими елементами) не повинна бути більшою за 10^{-6}—10^{-7}, в той час як частота помилкового приймання двійкового елемента (сигнали, що відповідають нулю чи одиниці) в більшості аналогових каналів, що зараз у нас діють, рідко буває меншою ніж 10^{-3}—10^{-4}. Докладне вивчення цього питання показує, що найбільш небезпечними бувають імпульсні шуми та короткочасні розриви зв'язку, хоча не тільки це знижує шумостійкість. Перші зумовлені появою випадковим чином імпульсів різних амплітуд, тривалості, часу виникнення. До цього призводять як суто технологічні причини, пов'язані з якістю побудови каналів, так і експлуатаційні. Наприклад, потік імпульсних шумів значно збільшується під час проведення профілактичних робіт на магістралі, в яку входить канал передавання даних. Поява короткочасних розривів зв'язку спричинюється роботою комутаційних пристроїв.

Значного поліпшення якості каналів, особливо при далекому зв'язку та використанні великих швидкостей передавання, досягають включенням спеціальних пристроїв, наприклад, регенеративних трансляторів і фазових коректорів (корекція фазочастотної характеристики (ФЧХ) реального дискретного каналу). Це значно поліпшує якість приймання, але практично веде до суттєвого збільшення матеріальних затрат і не дає можливості досягти потрібної шумозахищеності систем передавання тільки за рахунок цього. Тому в системах передавання даних велике значення надається методам підвищення достовірності приймання, не пов'язаним із підвищенням якості каналів, а заснованим на використанні надлишковості при кодуванні повідомлень. Завдяки цим методам можна здобути потрібну шумостійкість пристроїв передавання з меншими затратами, зробивши це за будь-якої інтенсивності шумів або при досить низькій якості каналу зв'язку.

Розв'язати вказане завдання можливо використовуючи так звані коректуючі коди, в яких, окрім інформаційної, є також надлишкова службова інформація, котра й допомагає або виявити помилку в прийнятому повідомленні, або, якщо треба, навіть виправити її. Зрозуміло, що при цьому тривалість передавання комбінації збільшується (збільшується кількість її двійкових розрядів), в будь-якому разі зростає обсяг сигналу, що призводить або до зменшення пропускної здатності, або потребує розширення смуги пропускання, якщо треба зберегти значення цього параметра.

Таким чином, із загальної кількості N можливих комбінацій n-розрядних двійкових кодів вибирають за певною ознакою підмножину N_0 дозволених. Такі N_0 комбінацій і є коректуючими кодами.

Розглянемо цей процес утворення коректуючих кодів для дуже простого випадку: маємо множину $N = 2^3$ трирозрядних двійкових кодів (у загальному випадку $N = 2^n$); побудуємо підмножину N_0 коректуючих кодів, які дають змогу виявляти спотворення одного розряду. Напишемо всі можливі трирозрядні коди: 000, 001, 010, 011, 100, 101, 110, 111. Якщо вибрати підкреслені (або навпаки), то саме й матимемо $N_0 = 4$ коректуючих кодів, у яких неправильне приймання тільки одного розряду буде обов'язково виявлене, оскільки в цьому випадку дозволена комбінація завжди переходить у недозволену. Зазначимо, що в разі двох або трьох спотворень виявлення неправильного приймання комбінації неможливе, оскільки дозволена комбінація в результаті таких спотворень знову переходить в підмножину дозволених.

Як же із записаних $N = 2^3 = 8$ можливих комбінацій вибрати такі, щоб чітко виявити пошкодження двох елементів? У цьому простому випадку відповідь така: це пара — 000 та 111, або ж 001 та 110 і т. д.

Отже, підмножина коректуючих кодів знизилася до рівня $N_0 = 2$, але тепер лише три спотворення виключають можливість їх розпізнання.

Проаналізувавши наведений приклад, доходимо висновку, що у випадку з виявленням одного спотворення замість $N = 8$ можливих комбінацій використовуємо тільки $N_0 = 4$ (тобто для передавання коректуючого коду беремо $n = 3$ розрядів, в той час як при $N_0 = 4$ могли б брати $n_0 = 2$, але код був би некоректуючим). Коли побудували код, що виявляв дві помилки, то $N_0 = 2$, тобто для передавання було б достатньо $n_0 = 1$. В останньому випадку можна не тільки розпізнавати дві помилки, а й одну навіть виправляти.

З цих прикладів видно, що чим більше надлишковість ($N > N_0$ або $n > n_0$), тим краще розпізнавати більшу кількість спотворень в коректуючих кодах.

Мінімальну кількість спотворень d, яка одну дозволену комбінацію коректуючого коду переводить в іншу дозволену, називають відстанню Хеммінга або кодовою відстанню. А коди, які дають змогу виявляти певну кількість елементарних спотворень, називають *коректуючими кодами* з виявленням спотворень і, відповідно, якщо є можливість виправити помилки — коректуючими кодами з виправленням помилок.

Якщо врахувати сказане вище, а також зробити більш глибокий аналіз цього питання, то можна довести справедливість таких співвідношень:

$$d \geq \Delta + 1, \tag{1.1}$$

$$d \geq 2\sigma + 1, \tag{1.2}$$

де Δ — кількість розпізнаних помилок; σ — кількість помилок, що виправляються для даного коректуючого коду з відстанню Хеммінга d.

Таким чином, як випливає з формул (1.1) та (1.2), для виправлення будь-якої кількості спотворень у відповідному коректуючому коді потрібна вдвоє більша кодова відстань ніж в аналогічному випадку лише для розпізнавання такої самої кількості елементарних помилок.

Отже, можливості коректуючих кодів значною мірою залежать від рівня економічних витрат, завдяки яким одержуємо захист інформації від спотворень (більша ніж у простого коду кількість розрядів, час передавання, затрати на специфічні завдання з кодування та декодування тощо).

Показників, що відображають і шумостійкість коректуючого коду, і надлишкові затрати, три — два кількісні й один якісний.

Рівень затрат характеризують коефіцієнтом надлишковості, який визначають так:

$$K_н = \frac{\log N - \log N_0}{\log N}. \tag{1.3}$$

Можливості протистояти спотворенням інформації оцінюють через коефіцієнт пізнання спотворень $K_п$:

$$K_п = \frac{L}{L + M}, \tag{1.4}$$

де L — кількість комбінацій, в яких спотворення розпізнані; M — кількість комбінацій, в яких спотворення не розпізнані при передаванні Q комбінацій коректуючого коду (Q — досить велике число).

Вираз (1.4) зручно подавати через імовірності пізнання спотворень $p_п$ та нерозпізнання $p_н$ та будь-якого спотворення коректуючого коду $p_Σ$. В цьому разі, якщо вважати, що $Q \to \infty$, то $L / Q = p_п$, а $(L + M) / Q = p_Σ$.

Тоді співвідношення (1.4) переходить у

$$K_п = \frac{p_п}{p_Σ}. \tag{1.5}$$

Два варіанти знаходження $K_п$ згідно зі співвідношенням (1.5) наводяться для того, щоб раціональніше обчислювати значення цих величин у кожному окремому випадкові.

Третій показник (якісний) — це простота реалізації та наочність сприйняття. Між іншим, той спосіб здобуття коректуючих кодів,

який ми розглянули, дав негативні результати з цього показника (особливо з огляду на те, що насправді використовують не трирозрядні коректуючі коди, а значно довші — від кількох одиниць розрядів до кількох десятків і навіть сотень).

1.1.2. Класифікація коректуючих кодів та ймовірність помилок

Коректуючі коди поділяють на дві великі групи — блочні та безперервні. Блочні — це такі коди, в яких довжина кодових комбінацій визначена. Безперервні коди характеризуються тим, що у комбінації немає початку і кінця, причому інформація надходить безперервно, а надлишкові розряди (надлишковість) створюються безперервно і певним чином розміщуються серед інформаційних. Якщо інформація відсутня, процес передавання не зупиняють, а надсилають замість інформаційних деякі службові спеціально генеровані дані.

Такі коди можуть мати високі показники з виправлення спотворень, але є ряд причин, через які вони використовуються дуже рідко.

Найбільше використовують блочні коректуючі коди, які можна поділити на рівномірні, що мають сталу кількість розрядів, та нерівномірні, в яких довжина комбінацій може змінюватись. Як ті, так і інші можуть бути роздільні чи нероздільні. В роздільних коректуючих кодах місця для інформаційних і службових розрядів фіксовані. У нероздільних — ні. Ті коди, що були вже побудовані, нероздільні рівномірні.

Рівномірні коди поділяють на систематичні та несистематичні. Систематичними називають такі коди, для яких порозрядна сума «за модулем 2» будь-якої кількості дозволених комбінацій знову дає дозволену комбінацію. Ті елементарні коди, що були побудовані, є саме несистематичними.

У практиці використовують переважно систематичні коди, для них добре розроблена теорія побудови і застосування. Серед них найпоширеніші циклічні коди, оскільки вони дуже ефективні за віддаленого передавання в каналах із інтенсивними шумами, причому мають місце групові пошкодження розрядів (від кількох одиниць до кількох десятків і навіть сотень пошкоджень у пакеті). При цьому порівняно з іншими коректуючими кодами значно зменшуються затрати на надлишковість та дуже спрощуються пристрої кодування-декодування (кодери-декодери).

Циклічні коди — це підмножина систематичних, в яких дозволено циклічне переміщення розрядів зліва направо чи навпаки (на будь-яку кількість розрядів). Циклічні коди найбільш ефективні при розпізнаванні спотворень, але можуть бути використані і для виправлення помилок.

Серед систематичних дістали застосування також коди Хеммінга, які виправляють одну помилку, та підсилені коди Хеммінга, що можуть виявляти до трьох помилок.

Так званий код з парним (або непарним) числом одиниць також належить до систематичних і має чи не найширше використання як при передаванні, так і при зберіганні інформації.

Використовують також ітеративні коректуючі коди, в яких кожен інформаційний розряд входить у не менш ніж дві захищені групи, причому кожний тип захисту може будуватись однотипно або різнотипно. Наприклад, якщо інформацію записати у вигляді матриці, то кожен розряд може мати окремий захист за рядками і за стовпчиками. Приміром, через парність одиниць в рядках і стовпчиках, чи за рядками — парність одиниць, а за стовпчиками — перевірки Хеммінга, чи циклічного коду і т. ін.

Такі коди мають високі шумозахисні властивості за відносно невеликих затрат на надлишковість і зазвичай не реалізуються з допомогою спеціальних пристроїв кодування та декодування, а це робиться програмно без ускладнень, якщо програміст вважатиме, що стандартного захисту від шумів замало при передаванні інформації в якихось конкретних умовах.

Щоб оцінити ймовірність різних типів спотворень коректуючих кодів розглянемо просту модель цього процесу, вважаючи, що коректуючий код має n розрядів, а спотворення елементів коду (розрядів) статистично незалежні та ймовірність елементарного спотворення дорівнює p. Тоді ймовірність правильного приймання одного розряду дорівнює $1-p$, а ймовірність того, що вся n-розрядна комбінація не зазнає спотворень, становить $(1-p)^n$. Отже, ймовірність будь-яких спотворень в комбінації буде:

$$p_{\Sigma} = 1-(1-p)^n. \qquad (1.6)$$

Для того щоб можна було користуватися формулою (1.5), що визначає коефіцієнт пізнавання спотворень $K_{\text{п}}$, потрібно вміти визначати значення величин $p_{\text{в}}$ і $p_{\text{п}}$ (імовірності пізнання та непізнання пошкоджень коректуючого коду). Для цього застосуємо відому формулу біноміального розподілу, яка в даному випадку дає можливість обчислити ймовірність того, що в кодовій комбінації із n розрядів буде рівно K спотворень (позначимо цю ймовірність як p_n^K):

$$p_n^K = C_n^K p^K (1-p)^{n-k}.$$

Така проста модель має місце в системах відносно недалекої дії. Що ж до каналів, по яких передають інформацію на далекі відстані через комутатори, то тут можливі шуми, які призводять до чіткого статистичного зв'язку між спотвореннями елементів. Наприклад,

при передаванні даних телефонними каналами зв'язку до появи спотворень їх імовірність відносно мала ($10^{-3}...10^{-4}$), але з виникненням першого ж спотворення ймовірність наступного набагато більша (буває близькою до одиниці) і т. д. Це спричинює групові (пакетні) спотворення інформації, причому підряд або з якимись паузами. Можуть бути десятки, а то й сотні спотворень.

Тут потрібна інша модель. Для спрощення цього питання і можливості побудови інженерних методів розрахунків потрібних величин користуються таким прийомом. Вважають, що пакети помилок статистично не пов'язані і їх імовірність p така, як імовірність появи першої помилки (наприклад, $10^{-3}...10^{-4}$). Пошкодження в пакеті також статистично не зв'язані, але ймовірність елементарного спотворення інша і значно вища (в кожному окремому випадкові ці значення можуть бути різними).

1.1.3. Елементарні коректуючі коди

Код із парним чи непарним числом одиниць

Найпоширенішим і найбільш простим коректуючим кодом можна вважати код із парним (непарним) числом одиниць. Будують його так: кожна кодова комбінація має стале число розрядів n, причому тільки один із них є допоміжним і визначається так, щоб кількість одиниць у всій n-розрядній комбінації була парною. Тобто $n = n_0 + 1$, де n_0 — число інформаційних розрядів. Наведемо приклади для $n = 9$.

Якщо інформаційні частини двох кодів мають вигляд: 10101011 та 10100011, то відповідні їм коректуючі коди будуть: $1^{*}10101011$ та $0^{*}10100011$ (зірочкою позначено допоміжні розряди). Визначимо основні характеристики цього коду.

Коефіцієнт надлишковості згідно з виразом (1.3) (враховуємо, що $N = 2^n$, а $N_0 = 2^{n_0}$) буде:

$$K_{н} = \frac{\log 2^{n} - \log 2^{n_0}}{\log 2^{n}} = \frac{n - n_0}{n} = 1 - \frac{n_0}{n}.$$

Тепер оцінимо коефіцієнт пізнання спотворень $K_{п}$ згідно з виразом (1.5). Врахуємо, що, приймаючи такий коректуючий код, підраховують число одиниць в ньому і визначають, чи воно парне. Якщо так, то вважають, що спотворень немає, якщо ні, то навпаки, і з'являється сигнал «Спотворення». З урахуванням цього можна стверджувати, що будь-яку непарну кількість спотворень можна

розпізнати завжди і, навпаки, парну кількість — ніколи. Тоді явище, коли розпізнаються спотворення, зводиться до того, що відбулося або одне спотворення, або три, або п'ять і т.д.

Відповідно ймовірність $p_{п}$ за формулою (1.5) розраховується так:

$$p_{п} = p_{п}^{1} + p_{п}^{3} + p_{п}^{5} + ... = C_{п}^{1} p(1-p)^{n-1} + C_{п}^{3} p^{3}(1-p)^{n-3} + C_{п}^{5} p^{5}(1-p)^{n-5} + ...$$

Якщо, наприклад, імовірність $p = 10^{-3}$ (а вона, як правило, нижча), то вже другий член ряду менший від першого на 5 — 6 порядків. Тому вважатимемо, що в даному випадку

$$p_{п} = C_{п}^{1} p(1-p)^{n-1} = np(1-p)^{n_0}.$$

Тоді згідно зі співвідношеннями (1.5) та (1.6) для $K_{п}$ будемо мати:

$$K_{п} = \frac{np(1-p)^{n_0}}{1-(1-p)^{n}}.$$

Третій показник (якісний) — простота реалізації та наочність для розглянутого коду очевидні.

Звернемо увагу на те, що код, побудований подібно до описаного, тільки з непарним числом одиниць, матиме ті самі показники, крім одного — він може розпізнавати абсолютну непрацездатність пристроїв. Наприклад, якщо зовсім не працює пристрій, не реалізуючи операцій зчитування інформації, то це явище розпізнається, а при коді з парним числом одиниць — ні.

Цей простий код широко використовують для автоматичної реєстрації помилок у різних пристроях, особливо в запам'ятовувальних, а також при передаванні даних на близькі та далекі відстані як допоміжний захист інформації, котру реалізують як програмно, так і апаратно.

Коди з постійною вагою

Під вагою коду розуміють кількість одиниць в ньому. Коди з постійною вагою — це блочні рівномірні коди (мають сталу кількість розрядів n) із фіксованою кількістю одиниць. Для прикладу розглянемо міжнародний код № 3, який має $n = 7$ розрядів та вагу, що дорівнює три.

Зрозуміло, що в даному випадку дозволені будь-які семизначні комбінації, в яких рівно три одиниці і чотири нулі. Наприклад, 1110000, 0101010 чи 1100010. Коли їх декодують, то обчислюють

вагу. Якщо вага не дорівнює три, то з'являється сигнал «Помилка». За такого підходу розпізнаються будь-які збої (один, два, три і т. д.), за винятком парних, причому скільки спотворень одиниць, стільки має бути спотворень нулів (так звані спотворення зміщення).

Розрахуємо основні параметри такого коду при $N = 2^7$. Кількість же дозволених комбінацій, вага яких три, дорівнює кількості варіантів розміщення трьох одиниць на семи позиціях або чотирьох нулів на тому самому числі позицій, тобто

$$N_0 = C_7^3 = C_7^4.$$

Тоді для коефіцієнта надлишковості $K_н$ одержимо:

$$K_в = \frac{7 - \log C_7^2}{7} \approx \frac{2}{7};$$

загалом за ваги k мали б:

$$K_в = \frac{n - \log C_n^k}{n}.$$

У даному випадку оцінювати коефіцієнт пізнавання помилок $K_п$ краще через імовірність нерозпізнання спотворень $p_н$ з формули (1.5).

Із нерозпізнаних спотворень найбільш імовірними є подвійні типу зміщення (збивається тільки один нуль і тільки одна одиниця), інші, наприклад подвійні зміщення, мають на кілька порядків меншу ймовірність. Тому з погляду практики можна вважати, що пізнаються будь-які спотворення кодової комбінації, за винятком подвійного зміщення.

Розрахуємо ймовірність $p_н$. Явище, що призводить до нерозпізнання помилки в даному випадку, полягає в тому, що в комбінації спотворюється тільки одна з трьох одиниць (ймовірність p_3^1) і один із чотирьох нулів (ймовірність p_4^1). Отже, $p_в$ можна обчислити так:

$$p_п = p_3^1 p_4^1 = 3p(1-p)^2 4p(1-p)^3 = 12p^2(1-p)^5. \qquad (1.7)$$

Загалом за ваги k маємо

$$p_н = p_K^1 p_{n-k}^1 = k(n-k)p^2(1-p)^{n-2}. \qquad (1.8)$$

Таким чином, використовуючи співвідношення (1.5), (1.7) та (1.8) для коефіцієнта розпізнання, будемо мати:

$$K_п = \frac{1-(1-p)^7 - 12p^2(1-p)^5}{1-(1-p)^7}.$$

Загалом відповідно

$$K_п = \frac{1-(1-p)^n - k(n-k)p^2(1-p)^{n-2}}{1-(1-p)^n}.$$

Даний код не може бути схарактеризований наглядністю і простотою, а його реалізація ускладнюється у разі збільшення кількості розрядів. Наглядність не характерна для коду, оскільки він належить до групи нероздільних (не має постійного місця для інформаційних і допоміжних розрядів). Для кодування та декодування потрібні складні пристрої або значні витрати, пов'язані з роботою відповідних програм.

Кореляційний код

Кореляційний код будують так: кожний інформаційний розряд подають через два, причому замість одиниці передають 10, а замість нуля — 01.

Наприклад, якщо інформаційна частина записується як 110010101, то кореляційний код буде: 10100101100110010. Це блочний рівномірний коректуючий код, Декодується він так. При послідовному прийманні його елементів кожна їх пара, що відповідає одному інформаційному розрядові, перевіряється на непарність одиниць (тобто в парі має бути одиниця й обов'язково одна). Якщо десь виявиться парність (дві одиниці або два нулі), то з'являється сигнал «Помилка».

З огляду на сказане вище оцінимо основні характеристики такого коректуючого коду.

Відносно коефіцієнта надлишковості можна сказати, що він становить 50 % (0,5), оскільки інформаційна частина має n_0 розрядів, а кореляційний код — удвоє більше, тобто $n = 2n_0$.

Щоб вирахувати коефіцієнт пізнання спотворень $K_п$, з'ясуємо, які з них розпізнаються. Із суті кодування і декодування випливає, що будуть виявлені будь-які спотворення за винятком спотворень двох розрядів в одній чи більше парах. У разі збою у будь-якій хоча б одній парі навіть одного розряду, незважаючи на стан решти з'явиться сигнал «Помилка».

Таким чином, ураховуючи те, що одночасне спотворення в двох, трьох і т. д. парах не приводить до його розпізнавання, а ймовірність цих ситуацій на кілька порядків нижча, ніж для спот-

ворень двох розрядів тільки в одній парі, можемо стверджувати таке: з погляду практики кореляційний код розпізнає всі спотворення крім випадків, коли відбувається збій двох розрядів лише в одній парі. Виходячи з цього можемо вважати, що модель, яка описує ймовірності спотворень в даному випадку, можна подати через n_0-розрядний код з імовірністю спотворень одного елемента (що відповідає одній парі розрядів) p^2. Звідси ймовірність нерозпізнання:

$$p_{н} = p^1_{n_0} = C^1_{n_0} p^2 (1-p^2)^{n_0-1} = n_0 p^2 (1-p^2)^{n_0-1}.$$

Відповідно коефіцієнт пізнання помилок $K_{п}$ матиме вигляд:

$$K_{п} = \frac{1-(1-p)^{2n_0} - N_0 p^2 (1-p^2)^{n_0-1}}{1-(1-p)^{2n_0}}.$$

Простота реалізації і наглядність цього коду очевидні. Але треба врахувати, що високі показники з розпізнавання помилок одержані за рахунок великої, 50 %-ї надлишковості, а це — значні економічні витрати.

Інверсний код

Інверсний код іноді називають кодом із повторенням, що випливає з суті його побудови, а саме: інформаційна частина, що має сталу кількість розрядів n_0, передається двічі. Перший раз обов'язково в позитиві (тобто такою, як вона є): розряди один за одним передаються в канал з підрахуванням кількості одиниць. Друга частина передається так само, якщо число одиниць в першій було парним, або інвертується, якщо навпаки. Наведемо приклади для двох кодів з інформаційною частиною $n_0 = 8$:

Інформаційний код	Інверсний код
10010101	10010101 10010101
10011101	01100010 10011101

Тобто, як і для кореляційного коду, кількість розрядів n подвоюється $n = 2^{n_0}$ і надлишковість залишається такою самою (тобто 50 %).

Розглянемо техніку декодування. Приймання відбувається так: першу частину передачі розряд за розрядом завжди приймають в позитиві і підраховують кількість одиниць в ній. З другою частиною передачі роблять те саме, якщо в прийнятій першій було парне число одиниць, а в противному разі її інвертують. Потім перша і

друга частини складаються порозрядно «за модулем 2» і якщо тільки результат не нульовий, з'являється сигнал «Помилка».

Можна на прикладах упевнитися, що цей код достовірно (100 %) виявляє одне, два, три, п'ять, сім, дев'ять і т. д. спотворень. Розпізнає він також майже усі чотирикратні спотворення, крім випадку, коли два з них будуть на будь-яких місцях у першій частині і два — у другій, причому обов'язково на тих самих місцях, що й у першій частині. З погляду практики можна стверджувати, що тільки їх він і не розпізнає, бо шестикратні спотворення мають на кілька порядків менше значення ймовірності.

З огляду на сказане оцінимо ймовірність нерозпізнання помилок $p_{н}$. Явище, котре його спричинює, зводиться до того, що у першій частині із n_0 розрядів має бути тільки два спотворення й імовірність цього:

$$p_1 = p^2_{n_0} = C^2_{n_0} p^2 (1-p)^{n_0-2}.$$

Крім того, в другій частині має бути також два спотворення, але на тих самих позиціях з імовірністю p_2. Тоді $p_{н} = p_1 p_2$. Легко зрозуміти, що

$$p_2 = p^2 (1-p)^{n_0-2}$$

(ця величина менша за p_1).

Тоді

$$p_{н} = p_1 p_2 = C^2_{n_0} p^4 (1-p)^{2n_0-4}.$$

Слід звернути увагу на те, що для всіх розглянутих кодів не було такої низької ймовірності нерозпізнаних спотворень (вона пропорційна p^4).

Тепер коефіцієнт пізнання помилок

$$K_{п} = \frac{1-(1-p)^{2n_0} - C^2_n p^4 (1-p)^{2n_0-4}}{1-(1-p)^{2n_0}}.$$

З усіх розглянутих кодів за інших однакових умов (n_0 та p одні й ті самі) останній характеризується найбільшою шумозахищеністю навіть порівняно з кореляційним кодом, у якого така сама надлишковість. Цим підкреслюється те, що більша надлишковість дає більші потенційні можливості щодо розпізнання спотворень, але конкретні досягнення будуть залежати від того, наскільки ефективно чи не ефективно вона буде використана.

Оскільки розглянутий код дає можливість виявляти майже всі помилки (за прийнятої моделі ймовірних процесів), його у відповідних випадках використовують і для далеких передавань, тим більше, що зробивши детальний аналіз, можна було б упевнитися в непоганих його можливостях протидіяти і груповим (пакетним) помилкам.

Коди Хеммінга

Простий коректуючий код Хеммінга має кодову відстань $d = 3$ і дає змогу одне спотворення виправити і два розпізнати. Як і в решті випадків вводиться надлишковість через r службових розрядів, так що $n = n_0 + r$. Значення службових розрядів дістають за допомогою певних лінійних комбінацій з інформаційними (тобто якимсь чином сумуються «за модулем 2» певні інформаційні розряди).

У простому коді Хеммінга відомості про помилки одержують за допомогою r перевірок, що також являють собою певні суми «за модулем 2» його розрядів. Зрозуміло, що результат кожної з вказаних r перевірок може бути тільки нулем чи одиницею.

Ці перевірки і код Хеммінга будують так, що результати перевірок, записані в певному порядку, становлять двійковий код, який дає номер спотвореного розряду, якщо пошкодження було, або нуль, якщо пошкоджень не було.

Тепер вирішимо, скільки потрібно службових розрядів r за заданої кількості інформаційних n_0. Зазначимо, що кількість перевірок, тож і число розрядів у коді, з яких він побудований, дорівнює r. Найбільше число, яке можна записати в цьому r-розрядному коді, має бути таким, аби можна було записати номер будь-якого з $n = n_0 + r$ спотворених розрядів. Тобто

$$2^r - 1 \geq n + r. \qquad (1.9)$$

Для практичної роботи з формулою (1.9) не потрібно логарифмувати чи користуватися відповідними таблицями, як це рекомендується в деяких підручниках. Тут краще йти методом проб. Наприклад, нехай $n_0 = 7$. Спробуємо взяти $r = 3$. Тоді $2^3 - 1 = 7$, а кількість розрядів $n = n_0 + r$ дорівнювала б $n = 7 + 3 = 10$. Отже, співвідношення (1.9) не виконується, потрібно спробувати $r = 4$. У цьому випадку $15 \geq 11$ і співвідношення (1.9) виконується. Якщо б спочатку взяли $r = 5$, то з формули (1.9) побачили б велику надлишковість, а ще за одну пробу в бік зменшення дійшли б того самого висновку ($r = 4$).

Побудуємо простий код Хеммінга, який дає змогу виправляти одне спотворення. Для спрощення розглянемо цю побудову на прикладі коду, в якого вибрано $r = 4$. Тоді $n \leq 15$ і кількість інформаційних розрядів може сягати межі $n_0 \leq 11$. Якщо ж $r = 4$, то згідно зі сказаним раніше стільки ж має бути і перевірок. Позначимо ці перевірки так: b_1, b_2, b_3, b_4. Зміст їх ще невідомий, а можливі результати для кожної — це одиниця або нуль. Крім того, якщо перевірки побудовані правильно і на їх основі записано двійковий код $b_4b_3b_2b_1$, то він вказує на номер розряду, в якому є помилка, або дає нулі, якщо її немає.

Виходячи з цього побудуємо таблицю «побажань» для перевірок $b_1 \ldots b_4$ за умови, що код Хеммінга уже є, а спотворення можливе тільки в одного розряду (розряди цього коду α_1, α_2, α_3,... розмістимо в стовпчику α_i) (табл. 1.1).

Таблиця 1.1

α_i	b_4	b_3	b_2	b_1
α_1	0	0	0	1
α_2	0	0	1	0
α_3	0	0	1	1
α_4	0	1	0	0
α_5	0	1	0	1
α_6	0	1	1	0
α_7	0	1	1	1
α_8	1	0	0	0
α_9	1	0	0	1
α_{10}	1	0	1	0
α_{11}	1	0	1	1
α_{12}	1	1	0	0
α_{13}	1	1	0	1
α_{14}	1	1	1	0
α_{15}	1	1	1	1

Розглянемо перший рядок табл. 1.1 для першого розряду α_1. Його зміст виражає бажання, щоб перевірки $b_4b_3b_2b_1$ дали нуль, а b_1 — одиницю в разі спотворення першого розряду, оскільки в цьому разі $b_4b_3b_2b_1$ — двійковий код 0001, якому відповідає число 1 (тобто вказано номер спотвореного розряду). Так само, наприклад, для дев'ятого розряду бажано, щоб b_4 та b_1 дали одиниці, а b_3 та b_2 — нулі, оскільки в цьому випадку одержали б двійковий код 1001, що вказувало б на спотворення в дев'ятому розряді. Так можна висловити побажання для результатів будь-якого числа перевірок r.

У випадку, коли жоден із розрядів коду Хеммінга не постраждав, бажано, щоб усі перевірки давали $b_r = b_{r-1} = \ldots = b_1 = 0$.

Побажання легко здійснити, якщо код Хеммінга побудувати так, аби сума «за модулем 2» всіх його розрядів, проти яких стоять одиниці у стовпчику b_1 , давала б нуль. Так само для стовпчиків b_2, $b_3,\ldots,b_r$. Тобто

$$\left.\begin{aligned}
&\alpha_1 + \alpha_3 + \alpha_5 + \alpha_7 \ldots = 0;\\
&\alpha_2 + \alpha_3 + \alpha_6 + \alpha_7 \ldots = 0;\\
&\alpha_4 + \alpha_5 + \alpha_6 + \alpha_7 \ldots = 0;\\
&\alpha_8 + \alpha_9 + \alpha_{10} + \alpha_{11} \ldots = 0.
\end{aligned}\right\} \qquad (1.10)$$

На основі системи рівнянь (1.10) можемо впевнитися, що всі побажання згідно з табл. 1.1 виконуються, якщо спотворюється тільки один розряд.

Виконати умови системи рівнянь (1.10) завжди можливо, оскільки тут ми маємо r рівнянь, які задовольняються відповідним підбором значень r службових розрядів у коді Хеммінга. При цьому номери службових розрядів не мають значення, але, щоб не ускладнювати питання, вибираємо такі, що входять тільки в одне з системи рівнянь (1.10). Це — α_1, α_2, α_4, α_8, α_{16},… і т. д.

Співвідношення, які використовуватимуться для визначення розрядних службових цифр коду Хеммінга, будуть записані так:

$$\left.\begin{aligned}
&\alpha_1 = \alpha_3 \oplus \alpha_5 \oplus \alpha_7 \oplus \alpha_9 \oplus \ldots;\\
&\alpha_2 = \alpha_3 \oplus \alpha_6 \oplus \alpha_7 \oplus \alpha_{10} \oplus \ldots;\\
&\alpha_4 = \alpha_5 \oplus \alpha_6 \oplus \alpha_7 \oplus \alpha_{12} \oplus \ldots;\\
&\alpha_8 = \alpha_9 \oplus \alpha_{10} \oplus \alpha_{11} \oplus \alpha_{12} \oplus \ldots;
\end{aligned}\right\}. \qquad (1.11)$$

А самі перевірки b_1, b_2, b_3,…, b_r на основі системи рівнянь (1.10) матимуть вигляд:

$$\left.\begin{aligned}
&b_1 = \alpha_1 \oplus \alpha_3 \oplus \alpha_5 \oplus \alpha_7 \oplus \alpha_9 \oplus \ldots;\\
&b_2 = \alpha_2 \oplus \alpha_3 \oplus \alpha_6 \oplus \alpha_7 \oplus \alpha_{10} \oplus \ldots;\\
&b_3 = \alpha_4 \oplus \alpha_5 \oplus \alpha_6 \oplus \alpha_7 \oplus \alpha_{12} \oplus \ldots;\\
&b_4 = \alpha_8 \oplus \alpha_9 \oplus \alpha_{10} \oplus \alpha_{11} \oplus \alpha_{12} \oplus \ldots;
\end{aligned}\right\}. \qquad (1.12)$$

Розглянемо приклад побудови конкретної комбінації коректуючого коду Хеммінга, якщо інформаційна частина задається у вигляді коду 100101. Отже, $n_0 = 6$, а на основі співвідношення (1.9) r має дорівнювати чотири і, таким чином, $n = 10$. Далі, записавши у загальному вигляді код Хеммінга і помітивши зірочками позиції r допоміжних розрядів, розмістимо інформаційні розряди на відведених для них місцях:

$$\begin{array}{cccccccccc}
 & & \cdot & & & \cdot & & \cdot & \cdot & \\
a_{10} & a_9 & a_8 & a_7 & a_6 & a_5 & a_4 & a_3 & a_2 & a_1\\
1 & 0 & & 0 & 1 & 0 & & 1 & &
\end{array}$$

Через відомі інформаційні розряди $\alpha_3\alpha_5\alpha_8\alpha_7\alpha_9\alpha_{10}$ визначимо на основі системи рівнянь (1.11) значення допоміжних розрядів $\alpha_1\alpha_2\alpha_4\alpha_8$:

$$\alpha_1 = 1 \oplus 0 \oplus 0 \oplus 0 = 1;$$
$$\alpha_2 = 1 \oplus 1 \oplus 0 \oplus 1 = 1;$$
$$\alpha_4 = 0 \oplus 1 \oplus 0 = 1;$$
$$\alpha_8 = 0 \oplus 1 = 1.$$

Тепер код Хеммінга можна записати у вигляді: 1010101111.

Перевіримо, чи справді можна виправити помилку, якщо вона є, наприклад, у п'ятому розряді, тобто α_5, а не нуль, як у непошкодженому коді. Допустимо, що такий пошкоджений код одержано з каналу зв'язку і взагалі невідомо, чи є там помилка, чи її немає (є тільки впевненість, що більше однієї помилки там немає). Використаємо систему (1.12) для отримання значень перевірок $b_1 \ldots b_4$:

$$b_1 = 1 \oplus 1 \oplus 1 \oplus 0 \oplus 0 = 1;$$
$$b_2 = 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 0;$$
$$b_3 = 1 \oplus 1 \oplus 1 \oplus 0 = 1;$$
$$b_4 = 1 \oplus 0 \oplus 1 = 0.$$

Таким чином, перевірки $b_4b_3b_2b_1$ — це код 0101, що відповідає числу 5. Аналогічно можна впевнитися, що дві помилки завжди пізнаються (результат $b_4b_3b_2b_1$ не дорівнюватиме нулю). Зрозуміло, що в цьому випадку йдеться тільки про розпізнання факту пошкодження комбінації, а виправлення неможливе, бо такі дії могли б призвести до ще більшого спотворення коду.

Отже, розглянутий коректуючий код Хеммінга може виправляти одне спотворення і завжди виявляти його, якщо їх буде не більше ніж два.

Збільшуючи надлишковість, тож і кодову відстань, можна підсилити коректуючі можливості коду Хеммінга. Розглянемо при-

клад, що реалізує таку можливість. Якщо до n-розрядної кодової комбінації $\alpha_n\alpha_{n-1}\ldots\alpha_2\alpha_1$ добавити ще один розряд (наприклад, α_0) так, щоб загальне число одиниць було парним, то одержаний код $\alpha_n\alpha_{n-1}\ldots\alpha_2\alpha_1\alpha_0$ матиме збільшені коректуючі можливості.

Справді, якщо після приймання комбінації будуть здійснюватися не тільки перевірки $b_4b_3b_2b_1$, а й контролюватиметься парність одиниць, і коли помилок буде непарна кількість, то хоч один метод контролю дасть ненульовий результат.

Коди Хеммінга використовують головним чином всередині ЕОМ, наприклад, для захисту інформації в запам'ятовувальному пристрої (ЗП) великих і середніх машин, для контролю правильності записів на дисках і т. ін. При передаванні на великі відстані і значній імовірності великої кількості помилок (чотири і більше) такий код сам не може використовуватись, але як допоміжна складова захисту може реалізуватися програмно.

1.1.4. Циклічні коректуючі коди

Математична структура циклічних кодів

Циклічні коди є основним засобом боротьби з помилками при передаванні інформації по каналах, де діють пакетні (групові) спотворення розрядів. Це викликано тим, що високих якостей коду в боротьбі з груповими помилками досягають відносно малими затратами на надлишковість і дуже простими, порівняно з іншими кодами, пристроями для кодування і декодування.

Циклічні коди, як уже зазначалося при класифікації коректуючих кодів, — різновид систематичних і мають як ті самі якості, так і специфічні. Як будь-який коректуючий, циклічний код має n_0 інформаційних і r надлишкових (допоміжних чи перевірчих) розрядів, так що довжина комбінації становитиме $n = n_0 + r$.

Для дослідження циклічних кодів використовують спеціальний математичний апарат, так звану алгебру з остачею. Наведемо основні ідеї цього підходу в найпростішому вигляді.

Кожній кодовій n-розрядній комбінації $\alpha_n\alpha_{n-1}\ldots\alpha_2\alpha_1$ відповідає алгебричний поліном $G(x)$, степінь якого $n-1$, причому

$$G(x) = \alpha_n x^{n-1} \oplus \alpha_{n-1}x^{n-2} \oplus \ldots \oplus \alpha_2 x^1 \oplus \alpha_1 x^0,$$

де α_i — розрядні цифри коду (α_i може бути нулем або одиницею), x — фіктивна змінна.

Наприклад, кодовій комбінації 10110101 буде відповідати поліном $G(x) = x^7 + x^5 + x^4 + x^2 + 1$.

Зазначимо, що в поліномі $G(x)$ стільки членів, скільки одиниць у відповідному коді, тобто кількість членів дорівнює вазі коду.

Операції над поліномами та їхніми членами виконуються за правилами звичайної шкільної алгебри, за винятком того, що тут замість плюса (+) і мінуса (–) діє плюс «за модулем» ($\oplus$). Тобто там, де в звичайній алгебрі потрібно ставити мінус чи плюс, ставиться плюс «за модулем» $\oplus$, який означає, що коли підсумовується парне число одиниць (чи членів з однаковим степенем x), будемо мати нуль, а якщо підсумовується непарне — одиницю (або один x у відповідному степені).

У циклічних кодах дозволяються операції циклічного зміщення як уліво, так і вправо. Переміщення всіх розрядів коду на один крок (розряд) вперед еквівалентне множенню відповідного полінома $G(x)$ на x, а якщо розряди зміщувати на k позицій вперед, то це означає, що поліном $G(x)$ множиться на x^k. Наведемо приклад. Нехай маємо код 10011101. Йому відповідає поліном $G(x) = x^7 + x^4 + x^3 + x^2 + 1$. Нехай $k = 3$. Тоді $G(x)x^3 = x^{10} + x^7 + x^6 + x^5 + x^3$ і йому відповідає код 10011101000.

Для побудови конкретного циклічного коду вибирають деякий поліном $P(x)$ степеня $r = n - n_0$. Цей поліном $P(x)$ назвемо створювальним. Потім із усіх можливих поліномів, що мають степінь $n-1$, вибирають такі, що діляться без остачі на створювальний поліном $P(x)$. Відібрані таким чином поліноми відповідають дозволеним комбінаціям n-розрядного циклічного коду.

Отже, циклічні коди мають поліноми, які без остачі діляться на створювальний поліном. Якщо при діленні полінома прийнятого коду є остача, то це ознака того, що в кодовій комбінації сталися спотворення і її можна забракувати, видавши відповідний сигнал про повторення передачі.

Від вигляду створювального полінома $P(x)$ суттєво залежать якості циклічного коду і насамперед його можливості щодо розпізнавання певної кількості помилок. Основні підходи до вибору $P(x)$ будуть описані пізніше, а зараз розглянемо, як можна створювати множини циклічних кодів за заданого $P(x)$.

Проведемо деякі формальні операції, використавши поліном формаційної частини $G(x)$ степеня $n_0 - 1$ і створювальний поліном $P(x)$ степеня r : помножимо $G(x)$ на x' і поділимо добуток $x'G(x)$ на створювальний поліном $P(x)$. Тоді

$$\frac{x'G(x)}{P(x)} = Q(x) + \frac{R(x)}{P(x)}, \qquad (1.13)$$

де $Q(x)$ — ціла частина від ділення $X^rG(x)$ на $P(x)$ того самого степеня, що і $G(x)$, а $R(x)$ — остача, причому має степінь не вищий за $r-1$.

Рівняння (1.13) перепишемо так:

$$P(x)Q(x) = x^rG(x) + R(x). \qquad (1.14)$$

На основі рівняння (1.14) можна зробити такі висновки щодо способів побудови циклічних кодів.

Перший спосіб: поліном інформаційного коду множать на створювальний. Наприклад, інформаційна частина — 101101 і їй відповідає поліном $G(x) = x^5 + x^3 + x^2 + 1$. Нехай створювальний поліном буде $P(x) = x^3 + 1$. Тоді $G(x)P(x) = x^8 + x^6 + x^2 + 1$ і цьому поліному відповідає дев'ятирозрядний циклічний код 101000101. Зрозуміло, що це не роздільний код (невідомо, де інформаційні, а де — службові розряди). Цей недолік є причиною того, що такі коди не використовують на практиці.

Другий спосіб: інформаційні розряди зміщують на r розрядів уліво, а на r позицій, що залишилися, вписують код, що відповідає поліному остачі $R(x)$. Тут чітко фіксовані позиції інформаційних і службових розрядів, тобто код — роздільний, а тому з циклічних кодів тільки він і використовується для практичних цілей.

Наведемо приклад побудови коду, взявши інформаційну частину — 101001 і створювальний поліном $P(x) = x^3 + 1$. Тоді $x^rG(x) = x^3(x^5 + x^3 + 1) = x^8 + x^6 + x^3$. Розділимо одержаний поліном на $P(x) = x^3 + 1$, щоб дістати $R(x)$.

Матимемо: $x^5 + x^3 + x^2$.

Отже, $R(x) = x^2$ і йому відповідає код 100. Змістивши на три розряди вліво інформаційну чистину 101001, а на ті місця, що лишилися, записавши код остачі 100, отримаємо циклічний коректуючий код 10101100.

Циклічні коди, що побудовані таким чином, не потребують складних пристроїв для кодування і декодування, що поряд з іншими якостями зумовило дуже широке їх застосування на практиці.

Теореми шумостійкості циклічних кодів

Дослідимо коректуючі якості циклічних кодів при різних видах створювального полінома $P(x)$. Спочатку потрібно вивчити, за яких спотворень поліноми прийнятих комбінацій циклічного коду діляться чи не діляться на створювальний поліном $P(x)$: якщо після спотворень таке ділення не дає остачі, то спотворення розпізнане, в противному разі — нерозпізнане.

Введемо означення. Кодова комбінація, що має стільки ж розрядів, як і циклічний код, причому з одиницями в розрядах, які

мають ті самі номери, що й спотворені в циклічному коді (решта всі нулі), називається вектором помилок.

Легко показати, що саме сума «за модулем 2» неспотвореної комбінації і відповідного вектора помилок дає спотворену комбінацію. Наприклад, нехай передано якусь комбінацію циклічного коду 101001100, а завдяки дії шумів у каналах зв'язку були спотворені помічені розряди 101001100, тобто комбінацію прийнято у вигляді 100010110. Згідно з означенням вектор помилок буде (одиниці тільки в помічених розрядах) 001011010. Перевіримо:

циклічний код	101001100
вектор помилок	$\oplus$ 001011010
спотворена комбінація, що і треба було довести.	100010110

Дії в кодах можна зіставити з аналогічними через поліноми:

$$\overline{F}(x) = F(x) + \Pi(x), \qquad (1.15)$$

де $F(x)$ — поліном неспотвореного циклічного коду; $\overline{F}(x)$ — поліном спотвореного циклічного коду; $\Pi(x)$ — поліном вектора помилок.

Із рівняння (1.15) очевидний висновок: якщо поліном $F(x)$ без остачі ділиться на створювальний поліном $P(x)$, то і поліном вектора помилок $\Pi(x)$ обов'язково ділиться на нього ж, оскільки $F(x)$ завжди ділиться. І навпаки, якщо $\Pi(x)$ не ділиться без остачі на $P(x)$, то те саме можна сказати і про $F(x)$, тобто в цьому випадку помилки розпізнаються. Отже, досліджуючи подільність векторів помилок $\Pi(x)$ на створювальні поліноми, можна зробити висновок про шумостійкість циклічних кодів, побудованих на тих чи інших створювальних поліномах.

Розглянемо деякі теореми циклічних кодів.

Теорема 1. Циклічний код, створювальний поліном якого $P(x)$ має більше ніж одного члена, розпізнає наявність будь-якої одиничної помилки.

Д о в е д е н н я. Нехай спотворено i-й розряд. Тоді поліном вектора помилок $\Pi(x) = x^{i-1}$. Але, як відомо, ділення без остачі одночлена на $P(x)$, що має два або більше членів, неможливе.

Теорему доведено.

Теорема 2. Циклічний код, створений поліномом $P(x) = x + 1$, розпізнає наявність будь-якого непарного числа помилок.

Д о в е д е н н я. Якщо $F(x)$ — поліном неспотвореного циклічного коду, то $F(x) / P(x) = T(x)$, оскільки остачі в цьому випадку не буде. Звідси $F(x) = T(x) + xT(x)$. Зрозуміло, що поліноми $T(x)$ і $xT(x)$ мають одну й ту саму кількість членів, наприклад, по t_1. Крім того, в $T(x)$ і $xT(x)$ може бути по t_2 членів з однаковими степенями, які при підсумовуванні $T(x) + xT(x)$ скоротяться. Тому можна стверджувати, що поліном $T(x)$ завжди має $2t_1 - 2t_2 = 2(t_1 - t_2)$ членів, тобто парне число. Таким чином, це — код з парним числом одиниць, який вже розглядався, саме він і розпізнає наявність будь-якого непарного числа помилок.

Теорему доведено.

Теорема 3. Циклічний код, створений поліномом $P(x) = x^m + 1$, також розпізнає будь-яку непарне число помилок.

Д о в е д е н н я. Досить показати, що $x + 1$ входить в $x^m + 1$ як співмножник. Справді,

$$(x+1)(x^{m-1} + x^{m-2} + ... + x^2 + x + 1) = x^m + x^{m-1} + ... + x^2 + x + x^{m-1} + \\ + x^{m-2} + ... + x^2 + x + 1 = x^m + 1.$$

Теорему доведено.

Теорема 4. Циклічний код, створений поліномом $P(x)$ степеня r, розпізнає будь-які пакети помилок завдовжки $l \le r$.

Під пакетом (групою) помилок завдовжки l розуміють не менш як дві помилки в комбінації, причому відстань від першої до останньої становить l розрядів. Наприклад, якщо в кодових комбінаціях

1001110101 і 1100101001

помилки мають місце в позначених зірочками розрядах, то це означає, що і в першому, і в другому випадках наявні групові помилки, причому довжина пакетів однакова і $l = 6$.

Д о в е д е н н я. Нехай в коді є пакет помилок завдовжки r, причому він починається з першого розряду. Тоді поліном вектора помилок матиме вигляд: $\Pi(x) = x^{r-1} + \alpha_{r-1}x^{r-2} + ... + \alpha_2 x + 1$, де α_{r-1}, α_{r-2},..., α_3, α_2 — коефіцієнти, що дорівнюють нулю чи одиниці. Якщо ж пакет починається з i-го розряду, то відповідний поліном буде $\Pi(x) = x^{i-1}(x^{r-1} + \alpha_{r-1}x^{r-2} + ... + \alpha_2 x + 1)$. Оскільки x^{i-1} як одночлен на створювальний поліном $p(x)$, що завжди є неодночленом, не ділиться, то потрібно перевіряти на подільність поліном у дужках. Але в даному випадку цей поліном завжди даватиме остачу, бо його степінь нижчий ніж у $P(x)$ (тобто менший за r).

Теорему доведено.

Теорема 5. Циклічний код, створений поліномом степеня r, не розпізнав тільки $1 / 2^{r-1}$ частину пакетів помилок завдовжки $r + 1$.

Д о в е д е н н я. Вважатимемо, що будь-які пакети завдовжки $r + 1$ мають однакову ймовірність, що в принципі відповідає дійсності.

Як і раніше, запишемо поліном вектора помилок, який починається з i-го розряду:

$$\Pi(x) = x^{i-1}(x^r + \alpha_r x^{r-1} + ... + \alpha_2 x + 1).$$

З усіх варіантів поліномів степеня r (що в дужках) тільки один поділиться без остачі на $P(x)$, а саме той, який повністю збігається з $P(x)$. Варіантів же буде стільки, скільки кодових двійкових комбінацій типу $\alpha_r\alpha_{r-1}...\alpha_2$, тобто 2^{r-1}.

Таким чином, теорему доведено.

Теорема 6. Циклічний код, створений поліномом степеня r, не розпізнає тільки $1 / 2^r$ частину пакетів помилок, завдовжки більш ніж $r + 1$, тобто $r + 2$, $r + 3$ .... .

Д о в е д е н н я. Нехай довжина пакета $l \ge r + 2$. Тоді матимемо поліном пакета помилок у такому вигляді:

$$\Pi(x) = x^{i-1}(x^{l-1} + \alpha_{i-1}x^{l-1} + \alpha_2 x + 1).$$

Зазначимо, що $l - 1 \ge r$, тобто при ділепні полінома пошкільному («у стовпчик») до одержання остачі в частці буде хоча б один член.

Можна стверджувати, що через певну кількість кроків такого ділення дістанемо поліном типу $\alpha_{r-1}x^r + \alpha_r x^{r-1} + ... + \alpha_2 x + 1$, причому множина цих поліномів визначається множиною двійкових кодів $\alpha_{r+1}\alpha_r\alpha_{r-1}... \alpha_3\alpha_2$, яких буде 2^r. Але серед цієї множини тільки один поліном ділиться без остачі на $P(x)$, той, що повністю збігається з $P(x)$, тобто тільки в цьому випадку помилка не буде розпізнана. Отже, співвідношення шансів нерозпізнати помилку до розпізнати становить $1 / 2^r$.

Теорему доведено.

Теорема 7. Циклічний код, створений поліномом $R(x)$, розпізнає будь-які одиночні помилки, а також всі подвійні, якщо вибрати число його розрядів n так, щоб воно було менше або дорівнювало деякому q, причому q — найменше з чисел, при якому двочлен $x^q + 1$ без остачі ділиться на $P(x)$.

Д о в е д е н н я. Стосовно одиночних помилок, то на основі теореми 1 все доведено, адже $P(x)$ ніколи не беруть як одночлен.

Щодо подвійних помилок, то поліном вектора помилок в цьому випадку за умови, що вони сталися в i-му та j-му розрядах, причому $j > i$, можемо записати так: $x^i(x^{j-i} + 1)$. Поліном $\Pi(x)$ не поділиться на створювальний поліном $P(x)$, якщо не ділиться співмножник $x^{j-i} + 1$. А ділення без остачі неможливе, оскільки $j - i < n \le q$, а q — найменше число, за якого ділення $x^q + 1$ без остачі на P можливе.

Теорему доведено.

Важливість теореми 7 полягає в тому, що при передаванні інформації на значні відстані в дискретних каналах із пакетними помилками ймовірність подвійних помилок вища, ніж для пакетів різної довжини. Тому доцільно будувати такий циклічний код, що обов'язково розпізнає дві помилки.

Деякі наслідки теореми 7. Для коректуючих кодів з відстанню Хеммінга — $d = 3$ зв'язок між кількістю допоміжних розрядів r і загальною довжиною комбінації n такий:

$$2^r \ge n + 1.$$

Це співвідношення було розглянуто при вивченні кодів Хеммінга.

На основі теореми 7 для циклічних кодів неабиякий інтерес мають значення величини q, визначені як $q = 2^m + 1$, де m — ціле число. Причини такі: доведено, що в такому випадку двочлени $x^q + 1$ є найменшими загальними кратними для всіх без винятку незвідних поліномів степеня m (незвідний — коли в ньому зроблені всі скорочення і його не можна розкласти на множники). Крім того, двочлен $x^q + 1$ ділиться без остачі на $x + 1$. Таким чином, можна зробити висновок: для будь-якого m можна побудувати циклічний код завдовжки $n = 2^m + 1$ на основі незвідного створювального його полінома $P(x)$ степеня m, що буде розпізнавати всі одиночні та подвійні помилки.

Розглянемо приклад. Візьмемо створювальний поліном четвертого степеня $P(x) = x^4 + x + 1$ ($m = 4$ і $q = 2^4 - 1 = 15$). Отже, за кількості допоміжних розрядів 4 кількість розрядів не повинна перевищувати 15. А тому кількість інформаційних розрядів $n_0 \le 15 - 4 = 11$. Побудуємо циклічний код на основі інформаційної частини 10011001, якій відповідає поліном $G(x) = x^7 + x^4 + x^3 + 1$. Дістанемо

$$G(x)x^r = (x^7 + x^4 + x^3 + 1)x^4 = x^{11} + x^8 + x^7 + x^4.$$

Після відповідного ділення маємо: $x^7 + 1$. Одержимо залишок $R(x) = x + 1$, якому відповідає код 0011. Таким чином, циклічний код буде мати вигляд: 100110010011.

Вибір створювальних поліномів

Тип створювального полінома має велике значення, а вибір потрібно узгоджувати з довжиною його комбінації n, кількістю інформаційних n_0 або службових r-розрядів. Зрозуміло, що коректуючі коди потрібно будувати з мінімальною надлишковістю, але з певними наперед заданими коректуючими можливостями.

За кодової відстані $d = 3$ є точне співвідношення для n, n_0 та r, що має такий вигляд: $2^r \ge n + 1$ або $2^r \ge n_0 + r + 1$. Якщо n_0 на основі цих співвідношень взяти максимальним, то надлишковість буде мінімальною без порушення коректуючих можливостей.

Загалом для $d > 3$ таких точних співвідношень немає, але існують деякі приблизні оцінки. Наведемо для прикладу оцінку за Хеммінгом. Домовимося, що коректуючий код завдовжки n-розрядів з n_0 інформаційними та r службовими розрядами має виявляти σ помилок.

Тоді всі можливі комбінації з помилками від однієї до σ, які треба розпізнати, створюють для кожної з дозволених комбінацій підмножину M, причому

$$M = \sum_{i=1}^{\sigma} C_n^i.$$

Разом з однією неспотвореною комбінацією це матиме такий вигляд:

$$\sum_{i=1}^{\sigma} C_n^i + 1 = \sum_{i=0}^{\sigma} C_n^i.$$

Всього ж таких комбінацій серед загальної кількості можливих буде:

$$2^{n_0} \sum_{i=1}^{\sigma} C_n^i,$$

звідки очевидні співвідношення:

$$2^n > 2^{n_0} \sum_{i=0}^{\sigma} C_n^i \quad \text{та} \quad 2^{n-n_0} > \sum_{i=0}^{\sigma} C_n^i.$$

З останніх співвідношень та враховуючи, що $r = n - n_0$, одержимо:

$$r > \log \sum_{i=0}^{\sigma} C_n^i.$$

Підберемо та обґрунтуємо вибір створювальних поліномів. Кодами, що мають високі коректуючі можливості, є так звані коди

Боуза—Чоудхурі—Хоквінгема (БЧХ). В них довжину кодової комбінації вибирають з відомого співвідношення $n = 2^m + 1$, а створювальний поліном $P(x)$ знаходять як найменше спільне кратне (НСК) так званих мінімальних поліномів $\alpha_i(x)$, які розроблені математиками і зведені в спеціальні таблиці. Таким чином:

$$P(x) = \text{НСК}\{\alpha_1(x)\alpha_3(x)...\alpha_{d-2}(x)\}. \qquad (1.16)$$

Частину таблиці мінімальних незвідних поліномів наведено нижче (табл. 1.2).

Таблиця 1.2

i	$d_i(x)$ при m, що дорівнює				
	3	4	5	6	7
1	x^3+x+1	x^4+x+1	x^5+x^2+1	x^6+x^2+1	
2					
3		$x^4+x^3+x^2+x+1$	$x^5+x^4+x^3+x^2+1$	$x^6+x^4+x^2+x+1$	
5			$x^5+x^4+x^2+1$	x^6+x^5+x+1	
7				x^6+x^3+1	

Такі коди можна дістати тільки для непарних значень відстані Хеммінга d. Якщо потрібно одержати такі коди для парних значень d, то досить створювальний поліном $P(x)$ помножити на $(x + 1)$, що збільшить d на одиницю.

При виборі полінома $P(x)$ потрібно дотримуватись умови, щоб він входив як співмножник у двочлен $x^n + 1$, оскільки на основі теореми 7 будуть розпізнаватися подвійні помилки.

Розглянемо на прикладі, як побудувати код БЧХ, коли задано: $n = 31$ та $d = 5$. Враховуючи, що потрібно виконати умову $n = 2^m + 1$, знаходимо $m = 5$. З іншого боку, $d - 2 = 3$, тому останнім співмножником у рівнянні (1.16) буде $a_{d-2}(x) = a_3(x)$. А це означає, що для побудови полінома $P(x)$ за формулою (1.16) потрібно подивитись у графу, де $m = 5$ (див. табл. 1.2), і взяти звідти два поліноми, а саме $a_1(x)$; $a_3(x)$ (за порядком зверху вниз). Отже:

$$a_1(x) = x^5 + x^2 + 1;$$

$$a_3(x) = x^5 + x^4 + x^3 + x^2 + 1;$$

$$P(x) = a_1(x)a_3(x) = x^{10} + x^9 + x^8 + x^6 + x^5 + x^3 + 1.$$

Щоб збільшити кодову відстань до шести, достатньо помножити одержаний створювальний поліном на $(x + 1)$, що дасть:

$$P(x) = x^{11} + x^8 + x^7 + x^5 + x^4 + x^3 + 1.$$

Такий спосіб збільшення кодової відстані можна застосовувати для будь-якого систематичного коду з непарним d.

У зв'язку з тим, що циклічні коди дуже поширені в системах передавання даних різного призначення, є тенденція до їх стандартизації, що викликає потребу стандартизації і створювальних поліномів. Труднощі полягають у тому, що умови роботи систем передавання дуже різняться, як і вимоги до вірогідності переданого. Вирішення завдання стандартизації методів збільшення вірогідності переданого за допомогою коректуючих кодів дало б у ряді випадків багато корисного: можна було б комплектувати системи передавання деякими стандартними блоками, узгоджувати роботу різних пристроїв передавання даних. Міжнародний консультативний комітет з телефонії та телеграфії (МКТТ) рекомендував для підвищення вірогідності приймання інформації в середньошвидкісних системах використовувати коректуючі коди з комбінаціями завдовжки 260, 500, 980 розрядів. При цьому завжди брати створювальний поліном типу $P(x) = x^{16} + x^{12} + x^5 + 1$.

Йому відповідав код БЧХ з $d = 4$ (одержано при $d = 3 + 1$), $m = 15$ — один поліном $a_1(x)$ із таблиці мінімальних незвідних поліномів (див. табл. 1.2), який помножено на $x + 1$, щоб збільшити d на одиницю.

Багаторазові тривалі випробування коду БЧХ за різних умов роботи показали його високу ефективність. Наприклад, при використанні цього коду для передавання даних комутованими телефонними каналами загального призначення частота неправильного приймання восьмирозрядних комбінацій (байтів), якими кодуються символи, не перевищувала 10^{-6} за ймовірності спотворення одного двійкового елемента 10^{-3}.

Розглянемо деякі питання щодо довжини кодових комбінацій. Цілком слушно вибираючи число n враховувати, що джерелом і споживачем даних є комп'ютери, які обмінюються з іншими комп'ютерами або якимись зовнішніми пристроями словами, кратними байтам. Звідси можна зробити висновок, що довжина інформаційної частини циклічних коректуючих кодів має бути кратна байтам. З іншого боку, циклічні коректуючі коди не дають можливості вільно поводитися з кількістю інформаційних розрядів за заданої довжини ходової відстані (відстань Хеммінга) чи вибраній

кількості службових розрядів. Тому на практиці дуже часто скорочують довжину циклічних кодів за рахунок зменшення кількості інформаційних розрядів. Такі коди прийнято називати укороченими.

Коректуючі можливості укорочених циклічних кодів не гірші ніж у нормальних повних (неукорочених) циклічних кодів. Надлишковість вища, а техніка кодування і декодування не змінюється. Проте циклічне зміщення кодової комбінації укороченого коду не завжди дає дозволені комбінації, тому укорочені коди називають псевдоциклічними. Рекомендовані МКТТ довжини комбінації циклічного коду в 250, 500, 980 розрядів за одного й того самого створювального полінома $P(x) = x^{16} + x^{12} + x^5 + 1$ приводять до псевдоциклічних кодів, а в неукороченого повного коду мало б бути тільки інформаційних розрядів $n_0 = 2^{15} - 1 = 33000$.

Синтез кодерів і декодерів

Функції кодування та декодування легко вирішуються програмно або із застосуванням мікропроцесорної техніки. Проте в системах передавання даних використовують спеціалізовані набагато простіші пристрої. Пристрої, які на основі інформаційної частини створюють циклічний код, називаються кодерами, а ті, що його декодують, декодерами.

Щоб зрозуміти суть роботи кодерів і декодерів, розглянемо прості випадки побудови циклічних кодів, причому зіставимо алгебричні дії відповідним діям у кодах. Нехай маємо інформаційну чотирирозрядну частину n_0 у вигляді коду 1001 та створювальний поліном $P(x) = x^3 + x + 1$. Згідно з правилами одержання циклічних кодів будемо ділити $x^r G(x)$ на $P(x)$ для одержання остачі $R(x)$:

$$G(x) = 1 + x^3;$$

$$x^r G(x) = x^6 + x^3.$$

Остачі $R(x) = x^2 + x$ відповідає код 110, а циклічний код відповідно буде 1001110.

Виконаємо ділення в кодах $x^r G(x)$ на $P(x)$ для одержання остачі.

Звернемо увагу на те, що кожний крок при діленні в кодах пов'язаний з тим, що код створювального полінома 1011 виставляють проти старшого значущого розряду дільника. При цьому немає значення, чи був цей значущий розряд спочатку, чи з'явився в результаті подальших дій (після першого кроку одержали 10000, причому значущої одиниці на цій позиції у початковому коді 1001000 не було).

У наведеному прикладі повторили те, що робили з поліномами.

З коду 0110, який не ділиться далі на 1011, дістаємо остачу. Підкреслимо, що суть частки нас зовсім не цікавить, нам потрібно лише правильно одержати остачу.

З останнього прикладу бачимо, що для побудови кодера слід мати r-розрядний регістр зсуву (далі будемо називати r-регістром), однорозрядний суматор «за модулем 2» (називатимемо аналізатором) і деякі найпростіші схеми (рис. 1.1), які дають змогу реалізувати такий алгоритм.

Рис. 1.1

На аналізатор (позначено $\oplus$) синхронним зсувом розряд за розрядом від старших розрядів подається те, що записано в інформаційному n_0-розрядному регістрі (він також регістр зсуву) і в r-регістрі, який перед початком обнуляється (тому там усі нулі).

Після кожного такту зсуву (такт — зсув на один розряд) у випадку одиниці на виході аналізатора (одиниця буде, коли на один із входів іде нуль, а на інший — одиниця, чи навпаки) до того, що в r-регістрі, додається порозрядно «за модулем 2» код створювального полінома без старшого розряду.

Після n_0 тактів, коли скінчилась інформаційна частина, в r-регістрі залишиться потрібна остача.

На основі наведеного алгоритму (див. рис. 1.1) можна синтезувати кодер, спрощену схему якого подано на рис. 1.2.

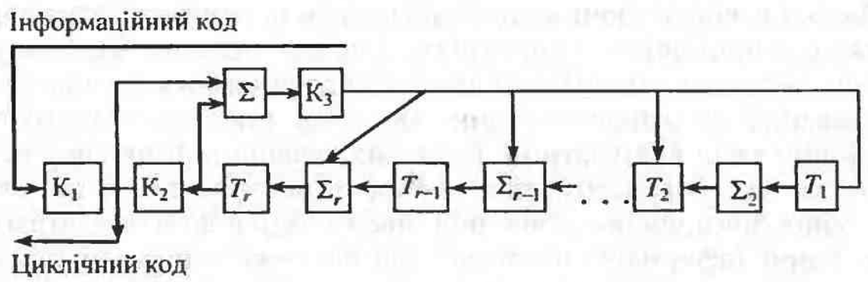


Рис. 1.2

До складу кодера входять:

1) r-розрядний регістр зсуву, що має деяку кількість суматорів «за модулем 2» (позначено — $\sum_2 ... \sum_r$), що дорівнює кількості одиниць в створювальному поліномі без однієї, причому суматори $\sum_i$ стоять на вході тих тригерів, у яких будуть одиниці, якщо записати код створювального полінома без старшого розряду;

2) три ключі $К_1$, $К_2$, $К_3$;

3) аналізатор на базі суматора «за модулем 2», з виходом на $К_3$.

Робота кодера. В початковому стані ключі $К_1$ і $К_3$ відкриті, а $К_2$ — закритий. Доки здійснюється n_0 тактів синхронного зміщення інформаційних розрядів, інформаційна частина через ключ $К_1$ розряд за розрядом виходить в канал, а також надходить на один із входів аналізатора (Σ). В цей час на аналізатор такт за тактом надходять також старші розряди r-регістра та згідно з описаним алгоритмом в r-регістрі формується остача. Через n_0 тактів ключі $К_1$ та $К_3$ закриваються, але відкривається ключ $К_2$ і з тією самою тактовою частотою одержана остача виходить у канал услід за інформаційною частиною.

Декодер має у своєму складі буферний регістр для прийому і запам'ятовування прийнятої інформації. Під час надходження з каналу перших n_0 розрядів вони такт за тактом ідуть в буферний регістр і кодер, а потім, починаючи з $n_0 + 1$-го такту, — тільки в кодер. Після n-го такту, коли закінчується приймання кодової комбінації, схема «або», на r входів якої (на рис. 1.2 не показана) подаються сигнали з кожного тригера, видає результат «помилка в», якщо r-регістр не обнулився (тобто коли спотворення циклічного коду привело до ненульової остачі при діленні комбінації на створювальний поліном).

Таким чином, у складі системи передавання даних, яка використовує циклічний код, і для передачі, і для приймання досить одного декодера, який буде робити в двох режимах — «Передавання» або «Приймання».

Коди, що виправляють помилки

Циклічні коректуючі коди розпізнають присутність спотворень розрядів у прийнятих комбінаціях, а це дає підставу підозрілу інформацію стирати й автоматично надсилати вимогу на повторне її передавання. В основному циклічні коди так і використовують. Але вони здатні вказувати не тільки на наявність помилок у комбінації, а й на номери розрядів, в яких є спотворення. Тоді для виправлення помилки потрібно лиш інвертувати відповідні розряди.

У теорії інформації показано, що систематичні коди (до яких належать і циклічні) можна будувати через так звану породжуючу матрицю, підсумовуючи відповідні її рядки «за модулем 2».

Для циклічних кодів така матриця може бути побудована дуже просто: достатньо взяти одиничну матрицю I розмірністю n_0 n_0 і допоміжну матрицю II розмірністю n_0 r, рядки якої — це відповідні остачі при побудові циклічних кодів від рядків одиничної матриці, що є інформаційними частинами циклічного коду:

$$n_0\left\{\underbrace{\left\|\overbrace{\text{I}}^{n_0}\ \overbrace{\text{II}}^{r}\right\|}_{n}\right. \text{ або } \left\|\begin{matrix} 100...0 & R_1(x) \\ 010...0 & R_2(x) \\ ..... & ..... \\ 000...1 & R_{n_0}(x) \end{matrix}\right\|.$$

Зрозуміло, що за заданої інформаційної частини, щоб одержати циклічний код, потрібно підсумувати «за модулем 2» ті рядки породжуючої матриці, що в сумі дають таку саму інформаційну частину.

Декодування можна вести не тільки розглянутим способом, а й на основі певним чином побудованих так званих перевірних матриць.

Один із способів побудови перевірних матриць для циклічних кодів використовує деякі $h(x)$ поліноми, які дістають так:

$$h(x) = \frac{x^n + 1}{p(x)}.$$

А сама перевірна матриця H має r рядків, що являють собою код, відповідний $h(x)$ і доповнений $r - 1$ нулями, тобто досить записати код для $h(x)$, додати $r - 1$ нуль, а решту рядків одержати циклічним зміщенням розрядів цього рядка:

$$H = \left\|\begin{matrix} \overbrace{00...0}^{r-1} & h(x) \\ 00...h(x) & 0 \\ ..... & ..... \\ h(x)\ 0... & 000 \end{matrix}\right\| \Big\} r.$$

Якщо матрицю H побудовано, то її рядки використовують як окремі перевірки, яких буде r. При цьому перевірні співвідношення полягають у тому, що у непошкодженого циклічного коду сума «за модулем 2» тих розрядів, номери яких збігаються з ненульовими значеннями відповідного рядка матриці H, має давати нуль. Якщо пошкоджень немає, то всі r перевірок, виконаних таким чи-

ном, повинні давати нулі. За умови, що елементами матриці H є h_{ij}, суть перевірок b_i загалом можна записати так:

$$b_1 = \sum_{j=1}^{n} h_{1j} a_j;$$

$$b_2 = \sum_{j=1}^{n} h_{2j} a_j;$$

.....

$$b_i = \sum_{j=1}^{n} h_{ij} a_j;$$

.....

$$b_r = \sum_{j=1}^{n} h_{rj} a_j.$$

Розглянемо простий приклад. Нехай потрібно побудувати циклічний код зі створювальним поліпомом $P(x) = x^3 + x + 1$, де $r = 3$, а $n = 2^3 - 1 = 7$. Знайдемо $h(x)$ як $(x^7 + 1) / (x^3 + x + 1)$. Відповідні розрахунки дають $h(x) = x^4 + x^2 + x + 1$ (зазначимо, що ділення завжди буде без остачі, оскільки на основі раніше сказаного $P(x)$ слід вибирати так, аби він входив співмножником у $x^n + 1$).

На основі одержаного $h(x)$ побудуємо перевірну матрицю H:

$$H = \begin{Vmatrix} 0\,0\,1\,0\,1\,1\,1 \\ 0\,1\,0\,1\,1\,1\,0 \\ 1\,0\,1\,1\,1\,0\,1 \\ 1\,2\,3\,4\,5\,6\,7 \end{Vmatrix}.$$

Тоді перевірні співвідношення b_i будуть записуватися так:

$$b_1 = a_3 \oplus a_5 \oplus a_6 \oplus a_7;$$
$$b_2 = a_2 \oplus a_4 \oplus a_5 \oplus a_6; \qquad (1.17)$$
$$b_3 = a_1 \oplus a_3 \oplus a_4 \oplus a_5.$$

Візьмемо будь-яку чотирирозрядну інформаційну частину, наприклад 1011, і побудуємо циклічний код при вибраному для даної матриці H створювальному поліномі $P(x) = x^3 + x + 1$. Це буде 1011000. Обчислимо результати перевірок на основі співвідношень (1.17):

$$b_1 = 0 \oplus 1 \oplus 0 \oplus 1 = 0;$$
$$b_2 = 0 \oplus 1 \oplus 1 \oplus 1 = 0;$$
$$b_3 = 0 \oplus 0 \oplus 1 \oplus 1 = 0.$$

Помилок немає.

Будувати перевірчі співвідношення було б значно легше, якби нумерація стовпчиків матриці H була така, як нумерація розрядів у циклічному коді (тобто справа наліво). Цього легко домогтись, якщо рядки перевірної матриці H записати в зворотному напрямі.

Виявляється, що перевірні співвідношення залишаються в силі, і ми маємо більш вигідну ситуацію для реалізації самих перевірок. Між іншим, побудована таким чином перевірна матриця H відповідає поліному $h(x)$ (код якого записують в зворотному напрямі щодо коду $h(x)$).

Згідно з наведеним раніше прикладом така матриця матиме вигляд:

$$H = \begin{Vmatrix} 1\,1\,1\,0\,1\,0\,0 \\ 0\,1\,1\,1\,0\,1\,0 \\ 0\,0\,1\,1\,1\,0\,1 \\ 7\,6\,5\,4\,3\,2\,1 \end{Vmatrix}, \qquad (1.18)$$

а відповідні перевірні співвідношення будуть записані як система (1.17).

У теорії інформації показано, що над рядками H дозволяються операції циклічного зміщення на будь-яку кількість розрядів як ліворуч, так і праворуч, а також підсумування «за модулем 2» будь-якої кількості рядків матриці H. Внаслідок таких операцій одержуємо знову рядки, на основі яких будуються перевірки.

З огляду на все сказане можна побудувати такі циклічні коди, які дають можливість не тільки розпізнати факт спотворень в прийнятій комбінації, а й виправляти певну кількість помилок. Один із способів виправлення — мажоритарне декодування, коли рішення про значення прийнятого розряду приймається за більшістю однотипних результатів непарного числа перевірок. Це можна зробити для циклічних кодів з непарною кодовою відстанню $d = \mu + 1$, де μ — парне число. Виходячи з того, що було сказано про кодову відстань раніше (див. формулу (1.2)), в цьому випадку можна розпізнавати до μ помилок або виправляти $\mu / 2$ помилок. Крім того, потрібно, щоб створювальний поліном $P(x)$ мав деякі специфічні властивості.

Розглянемо, в чому суть підходу. Побудуємо перевірну матрицю H на основі полінома $h(x)$. Як відомо, вона матиме r рядків та n стовпчиків:

$$H = \begin{Vmatrix} h_{11}\, h_{12} \ldots h_{1n} \\ h_{21}\, h_{22} \ldots h_{2n} \\ \ldots\ldots\ldots\ldots \\ h_{r1}\, h_{r2} \ldots h_{rn} \end{Vmatrix}.$$

За допомогою дозволених операцій підсумування рядків та їх циклічного зсуву побудуємо деяку іншу перевірну матрицю M, що буде мати μ (парна кількість) рядків (загалом $\mu \le r$):

$$M = \begin{Vmatrix} m_{11}\, m_{12} \ldots m_{1n} \\ m_{21}\, m_{22} \ldots m_{2n} \\ \ldots\ldots\ldots\ldots \\ m_{\mu 1}\, m_{\mu 2} \ldots m_{\mu n} \end{Vmatrix}.$$

При цьому матриця (1.18) повинна мати такі якості:

— тільки один із стовпчиків має всі одиниці;

— решта стовпчиків мають не більше однієї одиниці.

Згідно зі сказаним про перевірні матриці робимо висновок, що за допомогою матриці M можна зробити μ перевірок на парність для розряду, номер якого збігається з номером одиничного стовпчика в матриці (1.18). Додавши до цієї множини перевірок ще й очевидну $\alpha_i = \alpha_i$ (тобто враховуємо те, що в i-му розряді прийнято), одержимо $\mu + 1$ незалежних перевірних співвідношень для розряду α_i. Отже, матриця M має таку властивість, що кожний розряд, окрім α_i, входить в будь-яку з перевірок не більше одного разу. Така множина перевірок відносно розряду α_i називається *системою незалежних перевірок відносно* α_i.

Мажоритарне декодування проводиться так: якщо в прийнятій комбінації спотворень немає, то всі перевірки відносно α_i дадуть нульовий результат. Якщо сталося спотворення бодай одного розряду (в тому числі і того, що перевіряється), то одиничний результат отримаємо тільки в одній перевірці (кожен розряд входить в перевірки на основі матриці (1.18) лише один раз), якщо з помилками в двох розрядах — то тільки дві перевірки дадуть по одиниці і т. д.

Рішення відносно значення α_i приймають за однотипними результатами більшості таких перевірок (включаючи і перевірку $\alpha_i = \alpha_i$).

Таким чином, було перевірено тільки розряд α_i. Відносно інших розрядів, то синхронним пересуванням стовпців перевірної матриці (1.18) одиничний стовпчик можна поставити на будь-яку позицію. Можна діяти і по-іншому: циклічним зсувом прийнятої комбінації підставляти розряд, що перевіряється, під стовпчик матриці (1.18), в якому — всі одиниці.

Правильне декодування з виправлення помилок можливе тільки тоді, коли в комбінації циклічного ходу спотворено не більше $\mu / 2$ розрядів. За описаного методу декодування є можливість розпізнавати вдвоє більшу кількість помилок, а саме μ (в цьому випадку перевірки дадуть ненульовий результат).

Продемонструємо це на прикладі. Нехай треба побудувати декодуючий пристрій для циклічного коду БЧХ з даними $d = 5$; $P(x) = x^8 + x^7 + x^6 + x^4 + 1$ ($m = 4$ та $n = 2 - 1 = 15$). Теоретично при $d = 5$ можна виправляти дві помилки, отже, $\mu / 2 = 2$, тобто $\mu = 4$, а всього перевірок має бути (з урахуванням, що $\alpha_i = \alpha_i$) $\mu + 1 = 5$.

Отже, будуємо перевірну матрицю M типу (1.18), що матиме $\mu = 4$ рядків і $n = 15$ стовпчиків. Розділивши $x^n + 1$ на $P(x)$, тобто $x^{15} + 1$ на $x^8 + x^7 + x^6 + x^4 + 1$, одержимо $h(x) = x^7 + x^6 + x^4 + 1$ або $h(x) = x^7 + x^3 + x + 1$. Виходячи з $h(x)$ побудуємо матрицю H:

$$H = \begin{Vmatrix} 100010110000000 \\ 010001011000000 \\ 001000101100000 \\ 000100010110000 \\ 000010001011000 \\ 000001000101100 \\ 000000100010110 \\ 000000010001011 \end{Vmatrix}.$$

На основі цієї матриці або підсумуванням рядків, або циклічним переміщенням переробимо матрицю M типу (1.18) так, щоб одиниці були в крайньому правому стовпчику:

$$M = \begin{Vmatrix} 000000010001011 \\ 100000001000101 \\ 011000000010001 \\ 000101100000001 \end{Vmatrix}. \qquad (1.19)$$

Звідси для α_i можна записати такі перевірки:

$$\alpha_1 = \alpha_2 \oplus \alpha_4 \oplus \alpha_8;$$
$$\alpha_1 = \alpha_3 \oplus \alpha_7 \oplus \alpha_{11};$$
$$\alpha_1 = \alpha_5 \oplus \alpha_{13} \oplus \alpha_{14}; \qquad (1.20)$$
$$\alpha_1 = \alpha_9 \oplus \alpha_{10} \oplus \alpha_{12};$$
$$\alpha_1 = \alpha_1.$$

Перевірки для будь-якого іншого розряду можна одержати синхронним циклічним переміщенням стовпчиків матриці (1.19) так, щоб стовпчик з одиницями мав такий самий номер, що і розряд α_i, який перевіряється. В даному випадку на основі матриці (1.20) для перевірок α_i матимемо:

$$\alpha_i = \alpha_{i+1} \oplus \alpha_{i+3} \oplus \alpha_{i+7};$$
$$\alpha_i = \alpha_{i+2} \oplus \alpha_{i+6} \oplus \alpha_{i+10};$$
$$\alpha_i = \alpha_{i+4} \oplus \alpha_{i+12} \oplus \alpha_{i+13}; \qquad (1.21)$$
$$\alpha_i = \alpha_{i+8} \oplus \alpha_{i+9} \oplus \alpha_{i+10};$$
$$\alpha_i = \alpha_i.$$

Якщо згідно з перевірками (1.21) в правій частині будь-якої перевірки при підсумуванні індексів буде число $j > 15$, то, враховуючи циклічність зроблених переміщень, відповідний індекс визначають різницею $j - 15$.

Пересвідчимося на прикладі, що такий код дійсно виправляє помилки. Візьмемо інформаційну частину 0001110 і побудуємо циклічний код 000111010001000. Беремо будь-який, наприклад п'ятий, розряд і робимо перевірки за формулами (1.21):

$$\alpha_5 = \alpha_6 \oplus \alpha_8 \oplus \alpha_{12} = 0 \oplus 1 \oplus 1 = 0;$$
$$\alpha_5 = \alpha_7 \oplus \alpha_{11} \oplus \alpha_4 = 0 \oplus 1 \oplus 1 = 0; \qquad (1.22)$$
$$\alpha_5 = \alpha_9 \oplus \alpha_{22} \oplus \alpha_3 = 0 \oplus 0 \oplus 0 = 0;$$
$$\alpha_5 = \alpha_5 = 0.$$

Усі перевірки дали правильний результат.

А тепер нехай у п'ятому розряді буде помилка, тобто замість переданого нуля прийняли одиницю. Тоді всі перевірки дадуть нульовий результат, окрім п'ятої («сам на сам», тобто $\alpha_5 = \alpha_5 = 1$). Оскільки більшість (чотири) перевірок дали нуль, то і виправляємо одиницю на нуль.

Тепер припустимо, що з помилками восьмий та п'ятий розряди. Робимо перевірку для п'ятого розряду за (1.22). Друга, третя та четверта перевірки дадуть нуль, а перша та п'ята — одиницю. Рішення за більшістю — потрібен нуль (мажоритарний принцип).

Таким чином, зрозуміло, що декодування можна здійснювати тільки на основі матриці, де одиниці в якомусь одному стовпчику, наприклад, першому. Решта розрядів можуть бути перевірені відповідним циклічним переміщенням прийнятої комбінації так, щоб розряд, який перевіряється, був на першій позиції. Тоді реалізація декодуючого пристрою стане надзвичайно простою.

Зазначимо, що не всі циклічні коди дають можливість описаним способом будувати коди, за допомогою яких виправляються помилки, оскільки з перевірної матриці H, яку можна загасити для коду з будь-яким створювальним поліномом $P(x)$, не завжди вдається одержати матрицю M типу (1.18) з відповідними властивостями.

Поняття про ітеративні коди

Будь-яку кодову комбінацію з n розрядів, зокрема і комбінацію коректуючого коду, можна розглядати як n-мірний вектор. Тоді найпростіший приклад ітеративних кодів буде в тому разі, коли рядки матриці — вектори одного коректуючого коду, а стопці — іншого.

Позиції інформаційних і допоміжних розрядів показані на рис. 1.3, де *1*, *2*, *3* та *4* в загальному випадку — матриці, причому в матрицю *1* записано інформаційну частину, в матрицю *2* — перевірні розряди до кожного інформаційного рядка матриці *1*, в матрицю *3* — до кожного стовпчика матриці *1*, а в матрицю *4* — допоміжні розряди до стовпців матриці *2*.

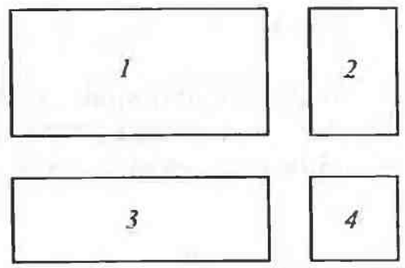


Рис. 1.3

Аналогічно цьому ітеративні коди можна будувати на основі тримірних таблиць та більш високих розмірностей, для яких графічних аналогів не існує.

Для ітеративних кодів, як доведено, справедливе співвідношення

$$d_{im} = \bigcup_{i=1}^{q} d_i,$$

де d_{im} — кодова відстань ітеративного коду; q — кількість ітерацій; d_i — кодова відстань для одного виду ітерації.

Для найпростішого випадку $q = 2$, тобто $d_{im} = d_1 d_2$. При цьому d_1 і d_2 — кодові відстані для рядків і стовпчиків.

Коефіцієнти надлишковості q-мірних ітеративних кодів розраховують так:

$$K_{н} = \frac{\prod_{i=1}^{q}(n_{0i} + r_i) - \prod_{i=1}^{q} n_{0i}}{\prod_{i=1}^{q}(n_{0i} + r_i)},$$

де n_{0i} та r_i — числа інформаційних та перевірних розрядів у рядку, стовпчику і т. д.

Ітеративні коди мають дуже великі захисні можливості, але внаслідок складності кодування та декодування використовують головним чином варіант із $q = 2$.

Розглянемо найпростіший варіант, коли перевірки за рядками та стовпчиками виконуються на парність числа одиниць.

$$\begin{Vmatrix} \alpha_{11}\ \alpha_{12}.....\alpha_{1n_{01}}\ \alpha_{1n_1} \\ \alpha_{21}\ \alpha_{22}.....\alpha_{2n_{01}}\ \alpha_{2n_1} \\ ..... \\ \alpha_{21}\ \alpha_{22}...\alpha_{n_2 n_{01}}\ \alpha_{n_2 n_1} \end{Vmatrix}.$$

У цій матриці α_{il} — значення двійкових розрядів; $n_1 = n_{01} + 1$ — довжина рядків; $n_2 = n_{02} + 1$ — кількість розрядів у стовпчику; n_{0i} та n_{02} — кількість інформаційних розрядів у рядку та стовпчику.

Таким чином,

$$K_{н} = \frac{n_1 n_2 - n_{01} n_{02}}{n_1 n_2} \approx \frac{n_{01} + n_{02}}{n_{01} n_{02}}.$$

У розглянутому коді $d = d_i^2 = 2^2 = 4$, тобто він здатен виявляти до трьох помилок, а також одну виправляти.

При декодуванні перевіряється парність одиниць в кожному рядку, а потім в кожному стовпчику. Якщо парності немає, то

з'являється сигнал «Помилка». Виправлення однієї помилки базується на тому, що в цьому випадку буде непарність одиниць в рядку і стовпчику, які перетинаються в місці спотворень розряду.

Легко переконатися, що одно-, дво-, три- й основна маса чотирикратних, усі п'ятикратні і т. д. помилки пізнаються. З практичного погляду розпізнається будь-яка кількість спотворень, за винятком деяких чотирикратних, а саме: коли в одному рядку є два спотворення, то стільки ж обов'язково на тих же позиціях буде в будь-якому іншому рядку.

Розрахуємо коефіцієнт розпізнавання помилок $K_{р}$. Ймовірність того, що нерозпізнана помилка буде в результаті відповідних двох спотворень в першому та другому рядках:

$$p_{1,2} = p_1 p_2 p_3 = C_{n_1}^2 p^2 (1-p)^{n_1-2} p^2 (1-p)^{n_1-2} (1-p)^{n_1(n_1-2)} =$$
$$= C_{n_1}^2 p^4 (1-p)^{2n_1-4} (1-p)^{n_1(n_1-2)} = C_{n_1}^2 p^4 (1-p)^{n_1 n_2 - 4},$$

де p_1 — ймовірність подвійної помилки в першому рядку; p_2 — те саме для двох пошкоджень па тих же позиціях в другому рядку; p_3 — ймовірність відсутності помилок в останніх $n_2 - 2$ рядках.

Відповідні ймовірності нерозпізнання помилок $p_{1,3}$, $p_{1,4}$ чи інших такого самого типу дорівнюють ймовірності $p_{1,2}$ і між собою, тому ймовірність нерозпізнання

$$p_{н} = C_{n_2}^2 p_{1,2} = C_{n_2}^2 C_{n_2}^2 p^4 (1-p)^{n_1 n_2 - 4}$$

і, відповідно,

$$K_{р} = \frac{p_{\Sigma} - p_{н}}{p_{\Sigma}} = \frac{1 - (1-p)^{n_1 n_2} - C_{n_1}^2 C_{n_2}^2 p^4 (1-p)^{n_1 n_2 - 4}}{1 - (1-p)^{n_1 n_2}}.$$

Розглянутий простий коректуючий код добре захищений від дії групових (пакетних) помилок. Він надійно (завжди) розпізнає пакети завдовжки до $l = n_1 + 1$ включно:

$$\alpha_{i1}\ \alpha_{i2}\ \alpha_{i3}\ \alpha_{i4}.....\alpha_{in_1};\ i\text{-рядок};$$

$$\alpha_{j1}\ \alpha_{j2}\ \alpha_{j3}\ \alpha_{j4}.....\alpha_{jn_1};\ j\text{-рядок},$$

де зірочками позначені спотворені розряди.

Як бачимо, пакет, який починається з α_{i3} і закінчується в α_{j3}, завжди розпізнається, а його мінімальна довжина дорівнює $n_1 + 1$ при тому, що i та j — сусідні рядки (тобто $j - i = 1$).

Якщо в проміжку між i та j є m непошкоджених рядків, то довжина розпізнаваного пакета буде $n_1(m + 1) + 1$. Пакети завдовжки $l = n_1 + 2$ в основному розпізнаються, але дуже рідко можуть і не розпізнаватися, коли рядки суміжні, у першому та другому є тільки по два спотворення, на тих самих позиціях. Такий коректуючий код широко використовують на практиці, причому кодування та декодування ведуть головним чином програмними методами.

Методичні рекомендації

Вивчаючи кодування інформації в каналах із шумом, слід з'ясувати собі напрями підвищення шумостійкості систем передавання даних та особливу роль і можливості надлишковості. Необхідно зрозуміти зв'язок між надлишковістю та можливостями покращення шумостійкості через коректуючі коди.

Для розуміння обґрунтованості вибору коректуючого коду потрібно знати, як оцінити його основні характеристики. Вивчаючи елементарні коректуючі коди, слід спробувати дати кожному окремому варіантові оцінку згідно з класифікацією.

Треба чітко розрізняти, які з кодів, що вивчаються, мають демонстративне, а які практичне значення.

Особливу увагу слід приділити вивченню циклічних кодів, зрозуміти, в чому їхня ефективність, що забезпечила їх поширення в практиці передавання на далекі відстані, особливо телефонними каналами зв'язку. Необхідно запам'ятати, як від типу створювального полінома залежить шумостійкість, як будуються коди, кодери та декодери. Певну увагу потрібно приділити вивченню мажоритарного способу декодування, циклічних кодів, що дає змогу виправляти помилки.

Необхідно освоїти техніку кодування та декодування найпростіших ітеративних коректуючих кодів, зрозуміти їхню високу шумостійкість, в тому числі і проти пакетних помилок, зробити висновки і запам'ятати, що це — один із важелів, який можна реалізувати програмно, якщо на практиці стандартні пристрої передачі даних не дають потрібного ефекту щодо захисту від помилок.

Питання для самоперевірки

1. Що таке коректуючий код?
2. Що таке відстань Хеммінга?
3. Як зв'язані коректуючі можливості з відстанню Хеммінга?
4. Які основні характеристики коректуючих кодів та як їх розрахувати?

5. Наведіть класифікацію коректуючих кодів.
6. Які помилки не розпізнаються в коді з парним числом одиниць і чому? Для інформаційних частин 10011101 та 110001001 побудуйте коди з парним числом одиниць.
7. Що таке коди з постійною вагою? Наведіть приклади кількох варіантів. Для коду з $n = 8$ та вагою 4 обчисліть коефіцієнт надлишковості та виведіть формулу для оцінки коефіцієнта розпізнання помилок.
8. Який коефіцієнт надлишковості у кореляційного коду? Як оцінити коефіцієнт шумостійкості за кількості інформаційних розрядів $n_0 = 8$ та ймовірності пошкодження одного розряду p?
9. Які помилки не розпізнає інверсний код? Чи відрізняються інверсний код і код із повторенням? При заданій імовірності спотворення одного розряду p запишіть ймовірність нерозпізнаних помилок.
10. Яка кодова відстань та скільки помилок може виправити та розпізнати простий код Хеммінга? Чи може він розпізнати наявність десяти помилок та 31 помилку?
11. Скільки помилок виправляє та скільки розпізнає підсилений код Хеммінга? Чи розпізнає він 22 помилки та п'ять?
12. Які комбінації в циклічному коді дозволені при заданому створювальному поліномі степеня r_i в інформаційній частині завдовжки n_0?
13. Як довідатися, чи є помилки в прийнятому циклічному коді?
14. Що таке вектор помилок? Запишіть вектор помилок для випадку, коли пошкоджені розряди помічені зірочками $\dot{1}0\dot{0}11\dot{0}10\dot{1}\dot{1}1$.
15. Що таке пакет помилок завдовжки l? Чи мають місце пакети помилок у наведених прикладах, а якщо мають, то які довжини пакетів:

$\dot{1}\ \dot{0}\ \dot{0}\ \dot{1}\ \dot{0}\ \dot{1}\ \dot{1}\ \dot{0}\ \dot{1}\ \dot{1}$; $1\ \dot{0}\ \dot{0}\ \dot{1}\ \dot{0}\ \dot{1}\ \dot{1}\ \dot{0}\ \dot{1}\ \dot{1}$; $1\ 0\ \dot{0}\ 1\ \dot{0}\ 1\ 1\ \dot{0}\ 1\ 1$;

$1\ 0\ 0\ 1\ 0\ 1\ 1\ \dot{0}\ 1\ 1$; $1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 1$.

16. Яким має бути створювальний поліном, щоб була можливість розпізнавати наявність непарного числа помилок?
17. Якщо створювальний поліном має степінь r, то якої довжини пакети помилок пізнаються?
18. Яка ймовірність пізнання пакетів помилок завдовжки r? $R + 1$? більше ніж $r + 1$?
19. Чи більша ймовірність пізнання пакетів завдовжки $r + 1$, ніж $r + 2$, $r + 3$ і т. д.? У скільки разів відрізняються відповідні ймовірності?
20. Що треба зробити, щоб у циклічному коді завжди розпізнавалися дві помилки?

21. Алгебричним способом та за допомогою послідовного алгоритму побудуйте циклічні коди, якщо створювальний поліном $P(x) = x^3 + x + 1$, а інформаційні частини такі: 1011, 1001, 1101, 1000.

22. Синтезуйте кодери для випадків, коли створювальні поліноми відповідно дорівнюють: $P_1(x) = x^3 + x + 1$; $P_2(x) = x^5 + x^2 + 1$; $P_3(x) = x^{10} + x^5 + x^4 + x^3 + 1$.

23. Чи можна побудувати циклічні коди, що виправляють помилки? В чому суть мажоритарного способу декодування?

24. Як у загальних рисах будується перевірна матриця та який зміст її рядків?

25. Що таке ітеративні коди? Як для них розрахувати основні характеристики?

26. Який найпростіший ітеративний код ви знаєте? Назвіть основні його властивості.

27. Чи стали поширеними ітеративні коди і в яких випадках їх доцільно використовувати?

1.2. Файлові системи й відновлення даних

1.2.1. Структури диска FAT

Жорсткий диск — основний пристрій для зберігання даних. Він може мати різну структуру зберігання файлів і каталогів, що забезпечує безпосереднє розташування даних на диску. Файлова система найчастіше інтегрована в операційну, а деякі операційні системи підтримують кілька файлових систем.

Більшість існуючих на сьогоднішній день файлових систем побудовані на основі **таблиці розміщення файлів** (File Allocation Table — FAT), що містить доріжки даних у кожному кластері на диску. Існує кілька типів файлової системи FAT — FAT 12, FAT 16 і FAT 32. Вони різняться кількістю цифр, використовуваних у таблиці розміщення файлів. Іншими словами, у файловій системі FAT 32 використається 32-розрядне число для зберігання доріжки даних у кожному кластері, в FAT 16 — 16-розрядне число й т. д. Сьогодні існують такі типи файлової системи FAT:

– FAT 12, використовувана в розділах ємністю не більше 16 Мбайт (наприклад, дискета);

– FAT 16, використовувана в розділах ємністю від 16 Мбайт до 2 Гбайт;

– FAT 32, використовувана (необов'язково) у розділах ємністю від 512 Мбайт до 2 Гбайт.

Файлові системи FAT 12 і FAT 16 завжди застосовували в операційних системах DOS і Windows. Вони підтримуються практично всіма відомими нині операційними системами. Більшість персональних комп'ютерів поставляється з жорсткими дисками, на яких установлена одна з файлових систем FAT.

Файлова система FAT 32 підтримується операційною системою Windows 95B і більш пізніми версіями, а також Windows 2000, що також підтримує файлову систему NTFS. Деякі операційні системи мають власну файлову систему. Наприклад, Windows NT і Windows 2000 підтримують файлову систему NT File System (NTFS); операційна система OS/2 поставляється із власною файловою системою High Performance File System (HPFS).

У цій главі йдеться про файлову систему FAT, а також розглядаються можливості нової системи FAT 32 і файлової системи NTFS.

Для забезпечення користувальницьким додаткам доступу до файлів незалежно від типу використовуваного диска в операційній системі передбачено кілька структур. Ці структури підтримуються операційними системами Windows 9X, Windows NT і Windows 2000 і представлені нижче в порядку розташування на диску:

- завантажувальні сектори головного й додаткового розділів;
- завантажувальний сектор логічного диска;
- таблиці розміщення файлів (FAT);
- кореневий каталог;
- область даних;
- циліндр для виконання діагностичних операцій читання / запису.

На відміну від жорсткого диска, на дискетах немає завантажувальних секторів головного й додаткового розділів і діагностичного циліндра. Ці структури створюються програмою Fdisk, що не застосовується для дискет, оскільки вони не можуть бути розбиті на розділи. На рис. 1.4 представлено взаємовідношення цих структур на диску *Western Digital Caviar* AC 12100 ємністю 2111 Мбайт.

Деякі знімні носії, наприклад *Iomega Zip*, функціонують подібно «високоємним дискетам», тобто на них немає завантажувальних секторів головного й додаткового розділів, а також діагностичного циліндра. Однак такі пристрої, як *Iomega Jaz*, схожі за структурою на жорсткі диски.

Кожна дискова область застосовується для конкретної мети. Ушкодження однієї з перелічених областей зазвичай призводить до

обмеження доступу до інших областей, викликаючи збої в роботі. Наприклад, операційна система не зможе одержати доступ до диска, якщо ушкоджено головний завантажувальний запис. Таким чином, розуміння логіки роботи кожної структури і їхньої взаємодії надає значну допомогу в усуненні неполадок.

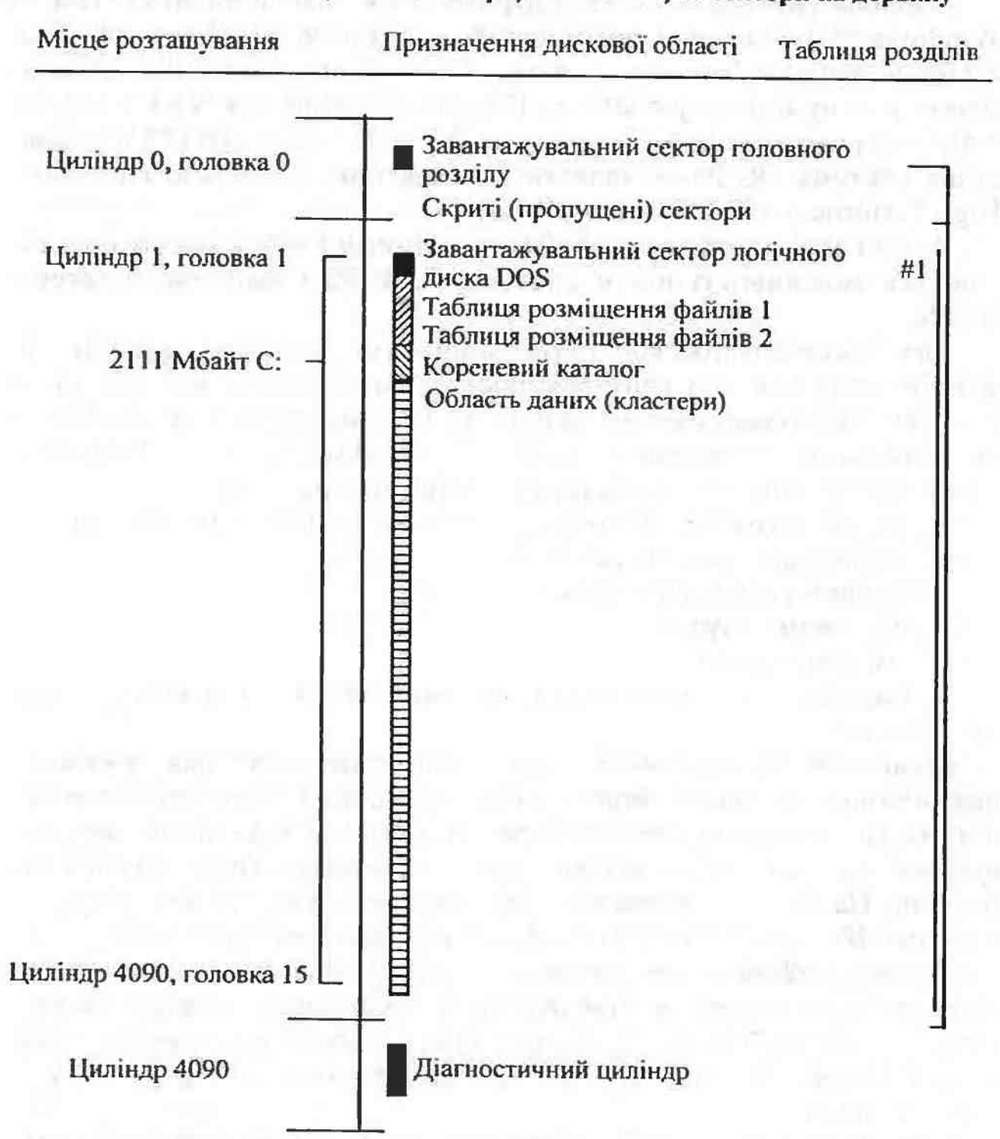


Рис. 1.4. Дискові системи DOS на диску Western Digital Caviar AC 12100

Завантажувальний сектор головного розділу

Уперше підтримка жорстких дисків була реалізована в DOS 2.0, випущеній у 1983 році. У цій операційній системі вперше застосовано 16-розрядну файлову систему FAT і підтримувався поділ диска на розділи, тобто створення на диску логічних томів. Виконувати це необхідно навіть у тому випадку, якщо ви збираєтеся використати тільки один розділ. Розділи диска іноді називають логічними томами, оскільки операційна система надає кожному розділові окрему букву.

Зараз практично всі накопичувачі на жорстких дисках діляться на кілька розділів, з якими працює встановлена операційна система. Поділивши диск, можна встановити кілька операційних систем — по одній в кожний розділ, і ці системи зможуть нормально співіснувати на одному диску.

Наприклад, ви можете за допомогою програми *F-disk* створити на диску один або кілька розділів для встановлення на них DOS або Windows 9X, а частину диска, що залишилася, надати для іншої операційної системи. Кожний розділ в операційній системі буде мати вигляд окремого диска.

Інформація про кожний розділ зберігається в завантажувальному секторі розділу (або логічного диска) на початку кожного розділу. Існує також основна таблиця списку розділів, поміщена в завантажувальний сектор головного розділу.

Завантажувальний сектор головного розділу (або *головний завантажувальний запис* (*Master Boot Record* — MBR)) є першим сектором на жорсткому диску (циліндр 0, голівка 0, сектор 1) і складається з двох елементів.

- *Таблиця головного розділу.* Містить список розділів на диску й розташування завантажувальних секторів відповідних логічних дисків. Ця таблиця дуже маленька й може містити максимум чотири записи. Таким чином, для одержання більшої кількості розділів в операційній системі (наприклад, DOS) можна створити один додатковий розділ і помістити в нього кілька логічних дисків.
- *Головний завантажувальний код.* Невелика програма, що виконується системою BIOS. Основна функція цього коду — передача керування в розділ, що позначений як активний (або завантажувальний).

Основні й додаткові розділи FAT

Кількість розділів на всіх жорстких дисках у системі може досягати 24. Це означає, що в комп'ютері може бути встановлено або 24 окремі накопичувачі, у кожному з яких є по одному розділу, або один жорсткий диск із 24 розділами, або кілька накопичувачів з різ-

ною кількістю розділів, але за умови, що загальна кількість розділів не більше 24. Якщо загальна кількість розділів перевищить цю цифру, DOS просто проігнорує їх, хоча інші операційні системи можуть працювати й з більшою кількістю томів. Таке обмеження DOS пов'язане з тим, що в латинському алфавіті від *C* до *Z* усього 24 букви.

На початку кожного розділу DOS є завантажувальний сектор логічного диска. При поділі диска на розділи необхідно створити активний (або завантажувальний) розділ. Програма, що міститься в найпершому секторі на жорсткому диску, визначає, який розділ активний, і передає керування його завантажувальному сектору. Ви також можете створити додатковий розділ диска для *Novell NetWare*, NTFS (Windows NT), HPFS (OS / 2), AIX (UNIX), XENIX або іншу файлову систему, використовуючи системний диск із відповідною програмою поділу диска.

Розділи, які використовуються цими операційними системами, недоступні при роботі в DOS. Вся справа в розходженнях між файловими структурами. DOS використає структуру FAT, що також підтримується OS / 2, Windows NT і деякими іншими операційними системами. У той же час в OS / 2 звичайно замість FAT застосовується файлова система HPFS (*High Performance File System*), а Windows NT користується власною файловою системою NTFS (NT *File System*) і т.д.

У табл. 1.3 наведено формат таблиці розділів, що зберігається в секторі головного завантажувального запису.

Таблиця 1.3

ГОЛОВНИЙ ЗАВАНТАЖУВАЛЬНИЙ ЗАПИС (таблиця розділів)

Зміщення	Довжина	Опис
	Перший запис у таблиці розділів	
1 BEh 446	1 байт	Індикатор завантаження (80h — активний, інакше 00h)
1 BFh 447	1 байт	Перша голівка (або сторона) розділу
1C0И448	16 біт	Перший циліндр (10 біт) і сектор (6 біт)
1C2И450	1 байт	Байт ідентифікації системи
1C3И451	1 байт	Остання голівка (сторона) розділу
1C4И452	16 біт	Останній циліндр (10 біт) і сектор (6 біт)
1C6И454	Одне подвійне слово	Відносне зміщення першого сектора на диску
1CAИ458	Одне подвійне слово	Кількість секторів у розділі

Продовження табл. 1.3

Зміщення	Довжина	Опис
	Другий запис у таблиці розділів	
1CEИ462	1 байт	Індикатор завантаження (80h — активний, інакше 00h)
1CFh463	1 байт	Перша голівка (або сторона) розділу
1 DOh 464	16 біт	Перший циліндр (10 біт) і сектор (6 біт)
1 D2h 466	1 байт	Байт ідентифікації системи
1D3h467	1 байт	Остання голівка (сторона) розділу
1 D4h 468	16 біт	Останній циліндр (10 біт) і сектор (6 біт)
1 D6h 470	Одне подвійне слово	Відносне зміщення першого сектора на диску
1 DAh 474	Одне подвійне слово	Кількість секторів у розділі
	Третій запис у таблиці розділів	
1DEh478	1 байт	Індикатор завантаження (80h — активний, інакше 00h)
1 DFh 479	1 байт	Перша голівка (або сторона) розділу
1EOh 480	16 біт	Перший циліндр (10 біт) і сектор (6 біт)
1 E2h 482	1 байт	Байт ідентифікації системи
1E3И483	1 байт	Остання голівка (сторона) розділу
1E4h 484	16 біт	Останній циліндр (10 біт) і сектор (6 біт)
1E6h 486	Одне подвійне слово	Відносне зміщення першого сектора на диску
1 EAh 490	Одне подвійне слово	Кількість секторів у розділі
	Четвертий запис у таблиці розділів	
1EEh 494	1 байт	Індикатор завантаження (80h — активний, інакше 00h)
1EFh 495	1 байт	Перша голівка (або сторона) розділу
1FOh 496	16 біт	Перший циліндр (10 біт) і сектор (6 біт)
1 F2h 498	1 байт	Байт ідентифікації системи
1F3h 499	1 байт	Остання голівка (сторона) розділу
1F4h 500	16 біт	Останній циліндр (10 біт) і сектор (6 біт)
1F6h 502	Одне подвійне слово	Відносне зміщення першого сектора на диску
1 FAh 506	Одне подвійне слово	Кількість секторів у розділі

Закінчення табл. 1.3

Зміщення	Довжина	Опис
Байти сигнатури		
1 FEh 510 2 байт		Сигнатура завантажувального сектора (55AAИ)

Слово відповідає двом байтам у зворотному порядку, подвійне слово — двом словам у зворотному порядку.

У табл. 1.4 наведено стандартні, а в табл. 1.5 — нестандартні значення байта ідентифікації системи.

Таблиця 1.4

БАЙТ ІДЕНТИФІКАЦІЇ СИСТЕМИ В ТАБЛИЦІ РОЗДІЛІВ (стандартні значення)

Значення	Тип розділу	Режим трансляції	Розмір розділу
00h	Немає	–	–
01 h	Основний, FAT 12	CHS	0-15 Мбайт
04h	Основний, FAT 1 6	CHS	16-32 Мбайт
05 h	Додатковий	CHS	16-32 Мбайт
06h	Основний, FAT 1 6	CHS	32 Мбайт-2 Гбайт
OEh	Основний, FAT 1 6	LBA	32 Мбайт-2 Гбайт
OFh	Додатковий	LBA	32 Мбайт-2 Гбайт
OBh	Основний, FAT 32	LBA	512 Мбайт-2 Гбайт
OCh	Додатковий	LBA	512 Мбайт-2 Гбайт

Таблиця 1.5

БАЙТ ІДЕНТИФІКАЦІЇ СИСТЕМИ В ТАБЛИЦІ РОЗДІЛІВ (нестандартні значення)

Значення	Тип розділу
02 h	Кореневий розділ MS-XENIX
03 h	Користувальницький розділ MS-XENIX
07h	Розділ файлової системи HPFS OS/2
08h	Розділ файлової системи AIX
09h	Завантажувальний розділ AIX
50h	Розділ Ontrack Disk Manager тільки для читання

Закінчення табл. 1.5

Значення	Тип розділу
51h	Розділ Ontrack Disk Manager для читання й запису
56h	Розділ Golden Bow Vfeature
61 h	Розділ Storage Dimensions Speedstor
63h	Розділ IBM 386/ix або UNIX System V/386
64h	Розділ Novell NetWare
75h	Розділ IBM PCIX
DBh	Розділ Digital Research Concurrent DOS/CPM-86
F2h	Другий розділ DOS версії 3.2+ (у деяких виробників)
FFh	Розділ дефектних блоків UNIX

При відновленні ушкодженого диска наведені в табл. 1.4 і 1.5 значення можна ввести за допомогою програми *Diskedit* з пакета *Norton Utilities*.

Не документовані можливості програми Fdisk

Fdisk — це програма зі значно більшими можливостями, які були поширені в DOS 5 і наступних версіях. На жаль, ці можливості ніколи не документувалися в посібнику з DOS і не були описані навіть в Windows. Найважливішим з не документованих параметрів є /mbr. З його допомогою програма *Fdisk* перезаписує дані в головному завантажувальному секторі, залишаючи незмінними таблиці розбиття. Параметр /mbr немов спеціально призначений для знищення вірусів, які «заражають» головний завантажувальний сектор диска (циліндр 0, голівка 0, сектор 1). Щоб скористатися цією можливістю, уведіть таку команду: **Fdisk / mbr**.

Послу цього *Fdisk* перезапише код завантажувального сектора, залишаючи таблиці розбиття незмінними. У нормально працюючій системі це не приведе до проблем, але про всяк випадок створіть резервну копію таблиць розбиття на дискеті.

Майте на увазі: таблиці розбиття будуть перезаписані в тому випадку, якщо 2 байти контрольного коду (сигнатури) 55AAh наприкінці сектора виявляться ушкодженими. Але ця ситуація малоймовірна. Насправді у випадку ушкодження байтів сигнатури ви зразу ж про це довідаєтеся: система перестане завантажуватися й буде поводитися так, начебто розбиття диска взагалі не існує.

Завантажувальний сектор

Завантажувальний сектор — це перший сектор на будь-якому логічному диску DOS. Наприклад, на дискеті або на диску *Zip* — це найперший фізичний сектор, оскільки дискету не можна розбити на розділи й вона має тільки один логічний диск. На жорсткому диску завантажувальний сектор (сектори) розташовується на початку кожного розділу (що не є додатковим) або на початку будь-якої області диска, розпізнаваної як логічний диск DOS.

Ці сектори трохи схожі на завантажувальні сектори розділів, тому що містять таблиці зі спеціальною інформацією про логічний диск.

- *Блок параметрів диска*, у якому міститься специфічна інформація, наприклад Розмір розділу, кількість використовуваних секторів диска, розмір кластера й мітка тому.
- *Завантажувальний код* — програма, що починає процес завантаження операційної системи. Для DOS і Windows *9x* це файл *Io.sys*.

Завантажувальний сектор дискети завантажується ROM BIOS, а при завантаженні системи з жорсткого диска MBR передає керування завантажувальному сектору активного розділу. В обох випадках завантажувальний сектор логічного диска дістає керування. Він виконує деякі перевірки й потім намагається прочитати з диска перший системний файл (в DOS / Windows це файл *Io.sys*). Завантажувальний сектор не видно, бо він перебуває поза областю зберігання файлів логічного диска.

Слід зауважити, що більшість сучасних систем підтримують завантаження з інших пристроїв, а не тільки з дискети. Ця можливість забезпечується системою BIOS. Наприклад, деякі системи можуть завантажуватися з накопичувача CD-ROM або диска *Zip* на додаток до завантаження з жорсткого диска й дискети.

Завантажувальний сектор логічного диска створюється програмою DOS і Windows *9x Format*. На жорсткому диску завантажувальні сектори є на початку кожного логічного диска як в основному, так і в додатковому розділах. Всі завантажувальні сектори поряд з даними про логічний диск містять спеціальний запис, однак при завантаженні виконується код тільки того сектора, що перебуває в активному розділі. Решта секторів просто зчитуються операційною системою для визначення параметрів логічних дисків.

Завантажувальний сектор логічного диска складається з програми (виконуваного коду) й області даних. Ця інформація необхідна операційній системі для визначення розміру логічного диска й розміщення таких структур як FAT. Формат блоку параметрів диска досить специфічний. Помилки в цьому блоці можуть призвести до проблем при завантаженні DOS або до відсутності доступу до диска.

У табл. 1.6 наведено формати завантажувального сектора DOS різних версій.

Таблиця 1.6

ФОРМАТИ ЗАВАНТАЖУВАЛЬНОЇ ЗАПИСУ РІЗНИХ ВЕРСІЙ DOS

Зміщення		Довжина поля	Опис
HEX	DEC		
00h	0	3 байти	Команда переходу на код завантаження
03 h	3	8 байтів	Ім'я виробника й версія DOS
0Bh	11	Одне слово	Розмір сектора в байтах (звичайно 512)
0Dh	13	1 байт	Розмір кластера в секторах (ступінь числа 2)
0Eh	/4	Одне слово	Кількість зарезервованих секторів (звичайно 1)
10h	16	1 байт	Кількість копій FAT (звичайно 2)
11h	17	Одне слово	Максимальна кількість записів у кореневому каталозі (звичайно 512)
13h	19	Одне слово	Усього секторів (якщо розділ не більше 32 Мбайт, у противному разі 0)
15h	21	1 байт	Байт опису диска (F8h для жорсткого диска)
16h	22	Одне слово	Розмір FAT у секторах
18h	24	Одне слово	Кількість секторів на доріжці
1Ah	26	Одне слово	Кількість голівок
1Ch	28	Одне подвійне слово	Кількість схованих секторів (якщо розділ не більше 32 Мбайт, тільки одне слово)
Для DOS версії 4.0 і більш пізніх, інакше 00h			
20h	32	Одне подвійне слово	Усього секторів (якщо розділ більше 32 Мбайт, інакше 0)
24h	36	1 байт	Фізичний номер диска (00h — дисковод, 80h — жорсткий диск)
25h	37	1 байт	Зарезервовано (00h)
26h	38	1 байт	Сигнатура розширеного завантажувального запису (29h)
27h	39	Одне подвійне слово	Серійний номер тому (розряди-32-розрядне випадкове число)
2Bh	43	11 байтів	Мітка тому ("NO NAME", якщо немає мітки)

Закінчення табл. 2.4

Зміщення		Довжина поля	Опис
HEX	DEC		
36h	54	8 байтів	Ідентифікатор файлової системи ("FAT12" або "FAT16")
Для всіх версій DOS			
3Eh	62	448 байтів	Код програми завантаження
1FEh	510	2 байти	Байти сигнатури (55ΛΑИ)

Слово відповідає двом байтам у зворотному порядку, подвійне слово — двом словам у зворотному порядку.

Каталог

Каталог — це база даних, що містить інформацію про записані на диску файли. Кожний запис у ній має довжину 32 байт, і між записами не повинно бути жодних розділювачів. У каталозі зберігається практично вся інформація про файл, який містить операційна система:

- *ім'я файла й розширення* — вісім символів імені й три символи розширення; крапка між ім'ям і розширенням файла мається на увазі, але не включається в цей запис (у Windows 9x ім'я файла може складатися з 255 символів у структурі каталогу);
- байт атрибутів файла, що містить прапорець, який представляє стандартні атрибути файла;
- час і дата створення файла або його модифікації;
- розмір файла в байтах;
- посилання на початковий кластер — номер кластера, з якого починається файл.

Інформація про розміщення файла, тобто розташування кластерів, що залишилися, міститься в FAT.

Існує два основні типи каталогів: *кореневий каталог* і *підкаталог*. Розрізняються вони максимальною кількістю файлів, що зберігаються. На кожному логічному дискові у фіксованому місці, зразу ж за копіями FAT, розташовується кореневий каталог. Розміри кореневих каталогів варіюються залежно від розміру диска, але кожний конкретний кореневий каталог має *фіксовану* максимальну кількість файлів. Довжина кореневого каталогу фіксується при створенні логічного диска й не може бути змінена в процесі роботи. Розмір кореневого каталогу різних накопичувачів наведено у табл. 1.7. На відміну від кореневого каталогу, підкаталог може зберігати довільну кількість файлів і розширюватися в міру необхідності.

Таблиця 1.7

РОЗМІР КОРЕНЕВОГО КАТАЛОГУ

Тип накопичувача	Максимальна кількість записів
Жорсткий диск	512
Дисковод 1,44 Мбайт	224
Дисковод 2,88 Мбайт	448
Jaz і Zip	512
LS-120	512
Sparq	512
Зауваження	
Одна з переваг FAT 32 полягає в тому, що кореневий каталог може бути в будь-якому місці диска й містити необмежену кількість записів.	

Усі каталоги мають однакову структуру. Записи в цій базі даних зберігають важливу інформацію про файли, пов'язану з інформацією, що зберігається в FAT, за допомогою одного з полів запису — номера першого зайнятого файлом кластера на диску. Якби всі файли на диску не перевищували розмірів одного кластера, потреби в FAT взагалі б не виникло. В FAT міститься інформація про файл, відсутня в каталозі, — номери кластерів, у яких розташований весь файл.

Для відстеження розташування всього файла на диску треба звернутися до каталогу й з'ясувати номер першого кластера й довжину файла. Потім, використовуючи таблицю розміщення файлів, переглядати *ланцюжок* зайнятих файлом кластерів, дійшовши до кінця файла.

Формат 32-байтового запису в каталозі наведено у табл. 1.8.

Таблиця 1.8

ФОРМАТ КАТАЛОГУ

Hex	Dec	Довжина поля	Опис
00h	0	8 байтів	Ім'я файла
08h	8	3 байти	Розширення файла
0Bh	11	1 байт	Атрибути файла
0Ch	12	10 байтів	Зарезервовано (OOh)
16h	22	Одне слово	Час створення

Закінчення табл. 1.8

Hex	Dec	Довжина поля	Опис
18h	24	Одне слово	Дата створення
1Ah	26	Одне слово	Початковий кластер
1Ch	28	Одне подвійне слово	Розмір файла в байтах

Слово відповідає двом байтам у зворотному порядку, подвійне слово — двом словам у зворотному порядку.

Імена файлів і їхніх розширень записані з прив'язкою до лівого краю й доповнені до максимальної довжини пробілами, тобто ім'я файла *AL* буде реально збережено як *AL*....., де крапки позначають пробіли. Перший байт імені файла також може позначати його стан, як наведено в табл. 1.9.

Таблиця 1.9

БАЙТ СТАНУ ЗАПИСУ КАТАЛОГУ (перший байт)

Hex	Стан файла
00h	Запис ніколи не використався, нижче цього запису пошук не виконується
05 h	Перший символ імені файла зараз — E5h
E5h	Файл вилучено
2 Eh	Крапка (.) показує, що запис є каталогом. Якщо й другий байт — 2Eh, то поле початкового кластера містить номер кластера батьківського каталогу (0000h, якщо батьківський каталог корсневий)

У табл. 1.10 наводяться використовувані в записах каталогів атрибути файлів.

Таблиця 1.10

АТРИБУТИ ФАЙЛІВ

Позиція біта в шістнадцятковому форматі								Значення	Опис
7	6	5	4	3	2	1	0		
0	0	0	0	0	0	0	1	01 h	Тільки для читання
0	0	0	0	0	0	1	0	02 h	Схований
0	0	0	0	0	1	0	0	04h	Системний
0	0	0	0	1	0	0	0	08h	Мітка тому
0	0	0	1	0	0	0	0	10h	Підкаталог
0	0	1	0	0	0	0	0	20h	Архівний (змінений)
0	1	0	0	0	0	0	0	40h	Зарезервовано
1	0	0	0	0	0	0	0	80h	Зарезервовано

Закінчення табл. 1.10

ПРИКЛАДИ

0	0	0	0	0	1	1	1	07h	Системний, схований, тільки для читання
0	0	1	0	0	0	0	1	21h	Тільки для читання, архівний
0	0	1	1	0	0	1	0	32h	Схований, підкаталог, архівний
0	0	1	0	0	1	1	1	27h	Тільки для читання, схований, системний, архівний

Таблиця розміщення файлів

Таблиця розміщення файлів (FAT) містить номери кластерів, у яких розташовані файли на диску. Кожному кластерові у FAT відповідає одне число. Сектори, що не містять користувальницьких даних (файлів), не відбиті в FAT. До таких секторів ставляться завантажувальні сектори, таблиці розміщення файлів і сектори кореневого каталогу.

У файловій системі FAT дисковий простір ділиться не на сектори, а на групи секторів, які називаються кластерами (*осередками розміщення*). Кластер містить один або кілька секторів. Розмір кластера визначається при поділі диска на розділи за допомогою програми *Fdisk* і залежить від розміру створюваного розділу. Найменший розмір диска, що може займати файл ненульового розміру, — один кластер. Кожний файл використає ціле число кластерів. Наприклад, якщо файл займає на один байт більше розміру кластера, то для його розміщення на диску буде виділено два кластери.

FAT — це електронна таблиця, що керує поділом дискового простору. Кожна комірка цієї таблиці пов'язана з певним кластером на диску. Число, що міститься в цій комірці, повідомляє про те, чи використаний даний кластер під який-небудь файл і, якщо так, то де перебуває наступний кластер цього файла.

Кожна комірка FAT зберігає 16-кове значення довжиною 12 або 16 біт. Шістнадцятирозрядні FAT більш зручні в роботі, оскільки значно легше редагувати поля розміром у два байти, ніж у півтора. Щоб самостійно відредагувати FAT, ви повинні виконати деякі математичні перетворення для одержання номера кластера. На щастя, багато програм дають змогу відредагувати FAT автоматично. Більшість із цих програм надає номери кластерів у десятковому вигляді, найзручнішому для користувачів. У табл. 1.11 наведено дані про каталог і FAT (файл нефрагментовано).

Таблиця 1.11

ЗАПИС ПРО НЕФРАГМЕНТОВАНИЙ ФАЙЛ У КАТАЛОЗІ FAT

Каталог		
Ім'я	Початковий кластер	Розмір
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	...
00999	0	Кластер доступний
01000	1001	Використається; посилання на наступний кластер
01001	1002	Використається; посилання на наступний кластер
01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	0	Кластер доступний
01005	0	Кластер доступний
...	...	...
65526	0	Останній доступний кластер

У даному прикладі запис каталогу вказує початковий кластер (1000), у якому міститься файл. У FAT кластери з ненульовими значеннями використаються, а спеціальне значення вказує подальше розташування файла. У розглянутому прикладі в кластері 1000 вказується кластер 1001, в 1001 — 1002, в 1002 — 1003, а в 1003 записане значення FFFFh, тобто на цьому кластері файл закінчується.

Розглянемо приклад із фрагментованим файлом. Нехай файл Usconst.txt записано, починаючи з кластера номер 1000. А файл Pledge.txt починається з кластера 1002. Таким чином, файл Usconst.txt стає фрагментованим. Описана ситуація ілюструється даними в табл. 1.12.

Таблиця 1.12

ЗАПИС ФРАГМЕНТОВАНОГО ФАЙЛА У КАТАЛОЗІ FAT

Ім'я	Початковий кластер	Розмір
Pledge.txt	1002	2
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	....
00999	0	Кластер доступний
01000	1001	Використається; посилання на наступний кластер
01001	1004	Використається; посилання на наступний кластер
01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	1005	Використається; посилання на наступний кластер
01005	FFFFh	Кінець файла
...	...	...
65526	0	Останній доступний кластер

У даному прикладі у файл Usconst.txt «укорінюється» файл Pledge.txt, що приводить до непослідовного розташування файлів на диску, тобто фрагментації. В операційних системах DOS і Windows є програми дефрагментації, які переміщають файли для їхнього послідовного розміщення на диску.

Перші два записи FAT зарезервовані й містять інформацію про саму FAT, решта вказують на відповідні кластери диска. Більшість записів FAT складається з посилань на кластери, в яких містяться частини певного файла, а деякі мають спеціальні шістнадцяткові значення:

- 0000h — кластер не використовується;
- FFF7h — як мінімум один сектор у кластері ушкоджений і не може бути використаний для зберігання даних;

■ FFF8h-FFFFh — кластер містить кінець файла.

Тип використовуваних FAT визначається програмою Fdisk, хоча записуються вони в процесі форматування високого рівня програмою Format. На всіх дискетах застосовується 12-розрядна FAT, а на жорсткому диску може використатися як 12-, так і 16-розрядна FAT, залежно від розміру логічного диска. На дисках розміром менше 16 Мбайт (32 768 секторів) застосовується 12-розрядна FAT, на дисках більшого розміру — 16-розрядна, а на дисках розміром більше 512 Мбайт при використанні Windows 95 OSR2 і Windows 98 — 32-розрядна FAT.

Програма *Fdisk* звичайно створює на одному диску дві копії FAT. Кожна копія займає кілька послідовних секторів на диску, і друга копія записується безпосередньо після першої. На жаль, операційна система використає другу копію FAT тільки в тому випадку, коли неможливо прочитати сектори, що містять першу копію. Таким чином, якщо перша копія FAT зникне (дуже поширена ситуація), операційна система не буде використати другу копію. Навіть команда *Chkdsk* не перевіряє другу копію FAT. Крім того, щораз, коли операційна система обновляє першу копію FAT, більші ділянки першої копії автоматично копіюються в другу. Якщо ж перша копія ушкоджена, то й друга копія виявиться ушкодженою: після відновлення FAT друга копія відбиває всі зміни в першій копії, включаючи й помилки. Обидві копії FAT рідко відрізняються одна від іншої, принаймні протягом тривалого строку: при відновленні перша копія FAT автоматично копіюється в другу. З огляду на все це, можна сказати, що застосування другої копії FAT обмежується тільки операціями з відновлення дефектних даних. Але навіть у такій ситуації використати другу копію FAT можна тільки тоді, коли проблема вирішується негайно, не чекаючи чергового відновлення FAT.

Кластер (осередок розміщення)

Термін *кластер* в DOS 4.0 був замінений терміном ***осередок розміщення*** (*allocation unit*). Новий термін — це синонім старого, тому що кластер є найменшим осередком на диску, яким може оперувати система при читанні або записуванні файла на диск. Кластер відповідає одному або (найчастіше) кільком секторам. Це дає змогу зменшити розмір FAT і прискорити роботу операційної системи, оскільки їй доводиться оперувати меншою кількістю розподілених осередків. У той же час зі збільшенням розміру кластера на диску росте й розмір дискового простору, що не використовується, бо його розподіл відбувається з дискретністю в один кластер.

У табл. 1.13 наведено стандартні розміри кластерів для різних форматів дискет.

Таблиця 1.13

СТАНДАРТНІ РОЗМІРИ КЛАСТЕРІВ ДЛЯ ДИСКЕТ

Тип диска	Розмір кластера (осередку розміщення)	Щільність
5,25-дюймовий на 360 Кбайт	Два сектори (1 024 байт)	Низька
5,25-дюймовий на 1,2 Мбайт	Один сектор (512 байт)	Висока
3,5-дюймовий на 720 Кбайт	Два сектори (1 024 байт)	Низька
3,5-дюймовий на 1,44 Мбайт	Один сектор (512 байт)	Висока
3,5-дюймовий на 2,88 Мбайт	Два сектори (1 024 байт)	Екстра

Досить дивною є та обставина, що деякі дискети високої щільності мають менший розмір кластера, ніж дискети низької щільності. Збільшується розмір FAT, зростає кількість записів, які повинна обробляти операційна система, і вповільнюється робота самої системи. Менший розмір кластера дає змогу зменшити розмір дискового простору, що не використовується. Весь простір між кінцем файла й кінцем останнього зайнятого кластера не використається, і в результаті, чим більше розмір кластера, тим більше втрати дискового простору. Крім того, дисководи високої щільності працюють швидше, ніж їхні «родичі» низької щільності. Все це дозволило IBM і *Microsoft* піти на зменшення розміру кластера в дискетах високої щільності, хоча при цьому й збільшується FAT.

Для жорстких дисків розмір кластера може варіюватися залежно від розміру розділу диска. У табл. 1.14 наведено розміри кластерів залежно від розміру логічного диска.

Таблиця 1.14

СТАНДАРТНІ РОЗМІРИ КЛАСТЕРІВ

Розмір диска, Мбайт	Розмір кластера	Тип FAT
Менше 16	8 секторів (4 096 байт)	12-розрядна
16-128	4 сектори (2 048 байт)	16-розрядна
128-256	8 секторів (4 096 байт)	16-розрядна
256-512	16 секторів (8 192 байт)	16-розрядна
512-1 024	32 сектори (16 384 байт)	16-розрядна
1 024-2 048 і більше	64 сектори (32 768 байт)	16-розрядна

Використання кластерів більших розмірів відчутно позначається на роботі системи. Наприклад, на диску ємністю 2 Гбайт, що містить 5 000 файлів, із середньою втратою дискового простору в півкластера на один файл сумарні втрати дискового простору становитимуть близько 78 Мбайт [5000×(0,5×32)].

Розмір кластера й структура FAT визначають максимально можливий Розмір розділу. Оскільки FAT використає записи розміром 16 байт для посилання на кластер у розділі, максимально можлива кількість кластерів може становити 65 536 (2). Максимальний розмір кластера — 32 Кбайт, отже, максимально можливий Розмір розділу — 2 047,6875 Мбайт.

Операційні системи Windows 95 0SR2jc і Windows 98 підтримують 32-розрядну FAT з розміром кластера до 64 Кбайт. З одного боку, завдяки використанню більшої кількості маленьких кластерів можна зменшити втрати дискового простору, а з іншого боку – більші кластери необхідні для більших логічних дисків. Так, використання 32-розрядних FAT дає змогу перевищити існуючий на даний момент ліміт в 2 Гбайт для одного розділу до 2 Тбайт (2 048 Гбайт). Взагалі ж, межа в 2 Гбайт існує тільки для DOS; такі операційні системи, як Windows *9x* і Windows NT, давно вже її перебороли.

Область даних

Область даних диска — це область, що випливає за завантажувальним сектором, таблицями розміщення файлів і кореневим каталогом на будь-якому логічному диску. Ця область контролюється за допомогою FAT і кореневого каталогу й ділиться на осередки розміщення, називані кластерами. У цих кластерах і розташовуються файли, що зберігаються на диску.

Циліндр для діагностичних операцій читання й запису

Програма розбивки диска на розділи *Fdisk* завжди резервує останній циліндр жорсткого диска для виконання діагностичних операцій. Через це *Fdisk* указує меншу кількість циліндрів, аніж існує насправді. Операційна система не використає цей циліндр, оскільки він перебуває поза розділами.

На системах з дисковими інтерфейсами IDE, SCSI або ESDI контролер повинен виділити додаткову область після розділів для зберігання таблиці зіпсованих доріжок і запасних секторів. У цьому випадку різниця між фактичною кількістю циліндрів і тою, що показує Fdisk, буде ще більшою.

Область діагностики дає змогу виконувати тестування читання/запису жорсткого диска, не ушкоджуючи даних на диску. Програми форматування жорстких дисків на низькому рівні звичайно використають цей циліндр для тестування чергування диска або для зберігання необхідної під час форматування інформації.

1.2.2. Файлові системи й відновлення даних

1.2.2.1. VFAT і довгі імена файлів

В оригінальній операційній системі Windows 95 використається та сама файлова система, що й в DOS, але зі значними змінами. У Windows 95 підтримується файлова система FAT, переписана в 32-розрядний код і названа **віртуальною таблицею розміщення файлів** (*virtual file allocation table* — VFAT). VFAT використається разом із 32-розрядною програмою VCACHE (яка замінила 16-розрядну програму SMART Drive з DOS і Windows 3.1) для забезпечення кращої продуктивності файлової системи. Однак основне істотне поліпшення нової файлової системи — це підтримка довгих імен файлів. Системи DOS і Windows 3.1 обмежувалися стандартом вісім-крапка-три при іменуванні файлів, тому додавання підтримки довгих імен файлів було пріоритетним завданням, яке необхідно було вирішити розроблювачам Windows 95, тим більше що користувачі операційних систем *Macintosh* і OS / 2 уже щосили застосовували ці можливості. Таким чином, творці Windows 95 мали забезпечити зворотну сумісність, тобто необхідно було реалізувати у файловій системі всі нові властивості й, крім того, не «обділити» користувачів попередніх версій DOS і Windows. До речі, зворотна сумісність — одна з найпоширеніших проблем у світі персональних комп'ютерів.

У файловій системі VFAT файла або каталогу можна надавати ім'я довжиною до 255 символів (включаючи шлях до цього файла або каталогу). В Windows 95 від трисимвольного розширення не відмовились, оскільки в цій операційній системі (як і в попередніх версіях Windows) за допомогою розширення створюється асоціація типу файл-додаток. У довгих іменах файлів можна застосувати пробіли, а також символи + „; = [], які не можна було використати в стандартних (вісім-крапка-три) іменах файлів DOS.

При створенні довгого імені файла створюється його псевдонім, що задовольняє стандарт вісім-крапка-три. Файлова система VFAT в операційній системі Windows *9x* виконує це в такий спосіб.

1. Перші три символи після останньої крапки в довгому імені файла стають розширенням псевдоніма.

2. Перші шість символів довгого імені файла (за винятком пробілів, які ігноруються) перетворяться в символи верхнього регістра й стають першими шістьома символами стандартного імені файла. Неприпустимі в стандартному імені файла символи (+,;=[]) перетворяться в символи підкреслення.

3. VFAT додає символи ~1 (сьомий і восьмий) до псевдоніма імені файла. Якщо перші шість символів кількох файлів однакові, то для розв'язання конфліктів імен додаються символи ~2, ~3 і т.д.

Довгі імена файлів у Windows NT

Зверніть увагу, що у Windows NT псевдоніми імен файлів створюються інакше, ніж у Windows *9x*. Операційна система Windows NT використає для створення «короткого» імені файла перші шість припустимих символів довгого імені й, якщо створене ім'я унікальне, додає символи ~1. Якщо ж перші шість символів уже використаються іншим файлом, то додаються символи ~2. Для створення розширення Windows NT використає перші три припустимі символи після останньої крапки в довгому імені файла. Якщо після додання символів ~5 з'являється ще одне аналогічне «коротке» ім'я файла, то для створення наступних імен файлів використається такий алгоритм: довге ім'я файла перетвориться в чотири шістнадцяткові символи, які містяться після двох припустимих символів довгого імені й додаються символи ~5. Таким чином, в Windows NT закінчення ~5 з'являється в усіх псевдонімів файлів, а змінюються тільки шістнадцяткові значення.

VFAT зберігає псевдоніми довгих імен у полі стандартних імен файлів запису каталогу файлів. Таким чином, усі версії DOS і Windows можуть дістати доступ до файла під довгим ім'ям за допомогою його псевдоніма. Залишається ще одна проблема: як зберігати 255 символів імені файла у 32 байтах запису каталогу, адже кожний символ імені файла — це один байт? Модифікувати структуру запису каталогу не можна, оскільки тоді попередні версії DOS не зможуть використати її.

Розроблювачі файлової системи вирішили цю проблему в такий спосіб: були додані додаткові записи каталогу для зберігання довгих імен файлів. Щоб попередні версії DOS не ушкодили ці додаткові записи каталогу, VFAT установлює для них атрибути, які не можна використати для звичайного файла: тільки для читання, скритий, системний і мітки тому. Такі атрибути DOS ігнорує, а отже, довгі імена файлів залишаються «недоторканими».

Існує ще одна проблема, пов'язана з довгими іменами файлів: додатки, що не вміють працювати з довгими іменами файлів, при

відкритті файла з довгим ім'ям і його наступним збереженням записуватимуть псевдонім файла в додаткові записи каталогу, отже, довге ім'я файла буде втрачено.

Старі програми для роботи з диском, такі як *Norton Disk Doctor*, не можуть працювати з файловою системою VFAT. Ці програми ігнорують додаткові записи каталогу. Тому після «відновлення» диска за допомогою таких програм ви можете не знайти довгих імен файлів.

При використанні VFAT я рекомендую застосовувати дискові утиліти, які підтримують цю файлову систему. Windows *9x* містить необхідні програми для перевірки, відновлення, дефрагментації диска й резервного копіювання. До речі, при запуску старих дискових утиліт в Windows *9x* ви будете попереджені про можливі наслідки. Якщо необхідно використати довгі імена файлів зі старими програмами, установіть програму *Lfnbk*.exe з компакт-диску Windows *9x*. Ця програма відновлює довгі імена файлів, але тільки в тому разі, якщо структура каталогу не змінювалася.

Існує ще одна проблема з довгими іменами файлів, що полягає в такому. VFAT створює новий псевдонім щораз при створенні або копіюванні файла в новий каталог. Наприклад, файл Expenses-January98 *.doc* зберігається в папці під псевдонімом EXPENS~1.DOC. Якщо за допомогою програми Windows 9x *Explorer* скопіювати цей файл у папку, в якій уже існує файл Expenses-December97 *.doc* із псевдонімом EXPENS-1. DOC, то VFAT створить у цій папці для файла, що копіюється, новий псевдонім EXPENS-2 . DOC. При цьому користувач не буде сповіщений про такі «самоправності». Для програм, що підтримують довгі імена файлів, таке копіювання — не проблема: всі довгі імена файлів зберігаються. Якщо ж запустити додаток, що не підтримує довгих імен файлів, то, відкривши файл EXPENS~1.DOC, користувач виявить, що це файл *Expenses-December97.doc*, а не *Expenses-January98.doc*.

1.2.2.2. FAT 32

Коли розроблялася файлова система FAT, на жорсткі диски розміром 2 Гбайт можна було натрапити хіба що в науково-фантастичних романах. Нині практично всі системи нижнього рівня оснащуються жорстким диском не менш 2 Гбайт, а найчастіше — 6 або 8 Гбайт. При використанні стандартної FAT ви можете створити розділ розміром не більше 2 Гбайт. Це обмеження призводить до незручностей в організації файлів для користувачів більш жорстких дисків: у їхньому розпорядженні як мінімум три диски (диск розміром 6 Гбайт можна розбити на три диски по 2 Гбайт).

Для усунення цього обмеження фірма Microsoft запропонувала нову файлову систему з розширеними можливостями — *FAT 32*. Ця файлова система працює як стандартна FAT, але має відмінності в організації зберігання файлів. Крім того, *FAT 32* можна встановити за допомогою програми *Fdisk*, на відміну від VFAT, що є частиною *Vmm.vxd*. Файлова система *FAT 32* була вперше реалізована в операційній системі Windows 95 OEM *Service Release 2* (OSR2). Вона вбудована також і в Windows 98. Крім того, *Microsoft* планує включити підтримку *FAT 32* в Windows NT 2000.

Оскільки FAT 32 установлюється тільки за допомогою програми *Fdisk*, ви не зможете використати цю файлову систему на дискетах і дисках знімних пристроїв, наприклад *Iomega Zip*, які не мають файлової системи. Однак такі пристрої як *Iomega Jaz* мають структуру жорстких дисків, і на них можна встановлювати файлову систему FAT 32.

Основна перевага FAT 32 — це можливість використання 32-розрядних записів замість 16-розрядних, що приводить до збільшення кількості кластерів у розділі до 268 435 456 (замість 65 536, або 2). Це значення еквівалентно 2, а не 2, оскільки чотири біти з 32 зарезервовані для інших цілей.

При використанні *FAT 32* Розмір розділу може досягати 2 Тбайт (1 Тбайт дорівнює 1 024 Мбайт). Нова файлова система може мати 4 294 967 296 (2) кластерів розміром 512 байт, а розмір одиничного файла може становити 4 Гбайт.

Існує ще одна відмінність *FAT 32* від її попередниць — положення кореневого каталогу: він не займає фіксованого місця на диску, як у *FAT 16*. Кореневий каталог у *FAT 32* може розташовуватися в будь-якому місці розділу й мати будь-який розмір. Усунення обмежень записів кореневого каталогу забезпечує динамічна зміна розміру розділу *FAT 32*. Фірма *Microsoft* не реалізувала цю чудову властивість в операційних системах Windows *9x*, чим і скористалися незалежні розробники, такі як фірма Power-Quest, що створила програму *Partition Magic*.

Основний недолік файлової системи *FAT 32* — несумісність із попередніми версіями DOS і Windows 95. Ви не зможете завантажити попередню версію DOS або оригінальну Windows 95 з диска з FAT 32, крім того, розділ з FAT 32 буде недоступний цим системам в разі їхнього завантаження з іншого диска.

1.2.2.3. Розмір кластера FAT 32

Оскільки розділ FAT 32 має більшу кількість кластерів, аніж розділ FAT 16, розмір кластера зменшується. Використання меншого

кластера знижує втрати дискового простору. Наприклад, розділ розміром 2 Гбайт із 5 000 файлів у FAT 32 використає кластер розміром 4 Кбайт замість 32 Кбайт у FAT 16. Таке зменшення розміру кластера сприяє зниженню втрат дискового простору з 78 до 10 Мбайт.

Для порівняння FAT 16 і FAT 32 необхідно подивитись, як у цих файлових системах організоване зберігання даних. Номери кластерів у FAT 16 зберігаються у вигляді 16-розрядних записів (0000h-FFFFh). Максимальне значення FFFFh відповідає десятковому 65 536, але кілька значень зарезервовані для спеціальних цілей. Реальна кількість кластерів у FAT 16 міститься у діапазоні 0002h-FFF6h, або 2-65 526. Таким чином, для зберігання файлів використається 65 524 кластери. Типовий запис про файл у FAT 16 представлено в табл. 1.15.

Таблиця 1.15

ЗАПИС ФАЙЛІВ У ФАЙЛОВІЙ СИСТЕМІ FAT 16

Каталог		
Ім'я	Початковий кластер	Розмір
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	...
00999	0	Кластер доступний
01000	1001	Використається; посилання на наступний кластер
01001	1002	Використається; посилання на наступний кластер
01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	0	Кластер доступний
...	...	...
65526	0	Останній доступний кластер

У файловій системі FAT 32 кількість кластерів міститься у діапазоні 00000000h-FFFFFFFFh, або 0-4 294 967 295. Як і в FAT 16, верхні

й нижні кластери зарезервовані для спеціальних цілей і їхні номери містяться у діапазоні 00000002h- FFFFFFF6h, або 2-4 294 967 286. Таким чином, для зберігання файлів можна використати 4 294 967 284 кластери. Накопичувач на жорстких дисках поділений на більшу кількість кластерів, кожний з яких стає менше, завдяки чому знижуються втрати дискового простору. Приклад записів про файл у файловій системі FAT 32 наведено у табл. 1.16.

Таблиця 1.16

ЗАПИС ФАЙЛІВ У ФАЙЛОВІЙ СИСТЕМІ FAT 32

Каталог		
Ім'я	Початковий кластер	Розмір
Usconst.txt	1000	8
FAT 32		
Номер кластера	Значення	Призначення
0000000002	0	Перший доступний кластер
...	...	...
0000000999	0	Кластер доступний
0000001000	1001	Використається; посилання на наступний кластер
0000001001	1002	Використається; посилання на наступний кластер
Номер кластера	Значення	Призначення
0000001002	1003	Використається; посилання на наступний кластер
0000001003	1004	Використається; посилання на наступний кластер
0000001004	1005	Використається; посилання на наступний кластер
0000001005	1006	Використається; посилання на наступний кластер
0000001006	1007	Використається; посилання на наступний кластер
0000001007	FFFFFFFFh	Кінець файла
0000001008	0	Кластер доступний

Каталог		
Ім'я	Початковий кластер	Розмір
...	...	...
4294967286	0	Останній доступний кластер

У табл. 1.17 наведено розмір кластера при використанні файлової системи FAT 32 з різними розмірами розділів.

Таблиця 1.17

РОЗМІР КЛАСТЕРА ФАЙЛОВОЇ СИСТЕМИ FAT 32

Розмір розділу	Розмір кластера, байт
До 260 Мбайт	512
260 Мбайт-8 Гбайт	4 096
6-16 Гбайт	8 192
16-32 Гбайт	16 384
32 Гбайт-2 Тбайт	32 768

Зменшення розміру кластера сприяє збільшенню записів у FAT. Розділ розміром 2 Гбайт із FAT 32 використає 524 288 записів, у той час як аналогічний розділ з FAT 16 використає 65 536 записів. І отже, таблиця FAT 16 має розмір 128 Кбайт (65 536 записів × 16 біт = = 1 048 576 біт, або 131 072 байт, або 128 Кбайт), а таблиця FAT 32 – 2 Мбайт.

Розмір FAT істотно впливає на продуктивність файлової системи. У Windows *9x* модуль VCACHE намагається завантажити FAT в оперативну пам'ять для поліпшення продуктивності системи. Вибір кластера розміром 4 Кбайт на дисках ємністю до 8 Гбайт забезпечує компроміс між продуктивністю і розміром FAT в оперативній пам'яті.

Незважаючи на те, що розмір FAT у файловій системі FAT 32 практично у двадцять разів більший ніж у FAT 16, з'являється незначний (менше 5 %) приріст продуктивності FAT 32 в операційній системі Windows 9x. Це почасти досягається використанням у персональних комп'ютерах найсучасніших накопичувачів на жорстких дисках.

Дзеркальна копія файлової системи

Файлова система FAT 32 також використає переваги двох копій FAT у поділі диска. Як і в FAT 16, у FAT 32 перша копія є основною і періодично копіює дані в додаткову копію FAT. У FAT 32 з появою проблем з головною копією FAT система перемикається на додаткову копію, що стає головною. Крім цього, система перериває процес створення дзеркальної копії FAT для запобігання втратам даних.

Створення розділу FAT 32

Для створення розділу з FAT 32 в Windows 9x необхідно використати програму Fdisk у командному рядку так само, як при створенні розділу з FAT 16. Під час запуску цієї програми буде виконане тестування диска й, якщо його розмір перевищує 512 Мбайт, з'явиться таке повідомлення:

Комп'ютер має диск ємністю більше 512 МБ. Дана версія Windows включає підтримку більших дисків і дає змогу ефективніше використати місце на таких дисках, а також форматувати диски розміром більше 2 Гбайт як один диск.

Якщо включити підтримку більших дисків і створити на них новий диск, неможливо буде дістати доступ до нового диска з іншої операційної системи, включаючи деякі версії Windows 95 і Windows NT, а також більш ранні версії Windows і MS-DOS. Крім того, дискові службові програми, які не підтримують явно файлову систему FAT 32, не зможуть працювати з цим диском. Якщо збираєтеся звертатися до цього диска з інших операційних систем або більш старих службових програм, не включайте підтримку більших дисків.

Включити підтримку більших дисків (Y / N).....? [N]

Якщо ви відповісте на це запитання ствердно, всі розділи розміром більше 512 Мбайт будуть мати файлову систему FAT 32. Крім того, така відповідь потрібна для створення розділу розміром більше 2 Гбайт. Наступні вікна роботи програми *Fdisk* аналогічні вікнам попередніх версій цієї програми.

Програма *Fdisk* автоматично визначає розмір кластера на основі обраної файлової системи й розміру розділу. Однак існує не документований параметр команди *Format*, що дає змогу явно вказати розмір кластера: *Format* / Z : і, де *n* — розмір кластера в байтах, кратний 512. За допомогою цієї команди ви можете створити файлову систему з розміром кластера, меншим установленого за замовчуванням.

Перетворення FAT 16 у FAT 32

Операційна система Windows 95 OSR2 може створювати розділи з FAT 32 тільки на порожньому диску. Для перетворення розділу з

FAT 16 необхідно скопіювати дані на інший носій, видалити розділ з FAT 16 і створити новий розділ з FAT 32, а потім відновити всі дані. В операційній системі Windows 98 є програма-майстер для перетворення розділу в FAT 32 без втрати даних.

При запуску програма перетворення диска відображає інформацію про існуючі розділи й установлені файлові системи (рис. 1.5). Вам необхідно виділити диск і виконати всі операції майстра.

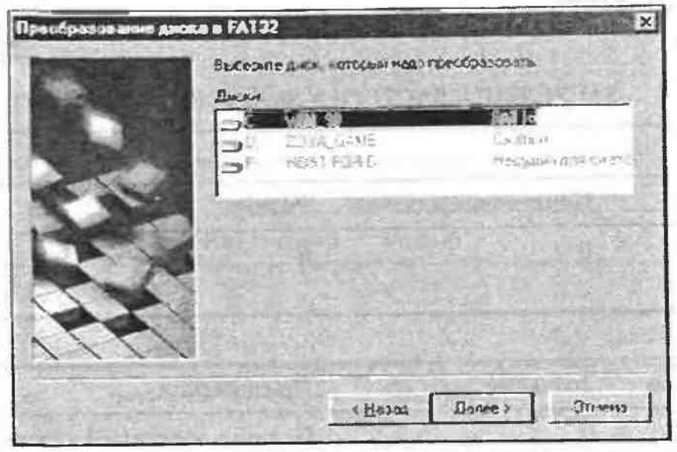


Рис. 1.5. Програма перетворення диска у FAT 32 з Windows 98

Зверніть увагу, що після перетворення диска в FAT 32 виконати зворотне перетворення не можна. Необхідно вживати «радикальних» заходів, тобто зберегти дані, запустити програму *Fdisk*, видалити розділ з FAT 32 і заново створити розділ з FAT 16.

FAT 32 п *PartitionMa&c*

Операційні системи Windows 95 OSR2 і Windows 98 містять базові інструменти для створення розділів з файловою системою FAT 32. Фірма *PowerQuest* створила програму *Parti-tionMagic*, що має неабиякі можливості роботи з файловими системами. За допомогою цієї програми ви можете виконати перетворення FAT 16 у FAT 32 і назад, а також змінити розміри розділів без утрати даних.

1.2.3. Помилки файлової системи FAT

Помилки у файловій системі стаються, найімовірніше, унаслідок програмних збоїв, аніж апаратних (наприклад, у разі неправильного завершення роботи Windows). Деякі програмні помилки описані далі.

1.2.3.1. Загублені кластери

Це найпоширеніша помилка файлової системи, за якої кластери у FAT позначаються як використовувані, хоча насправді такими не є. Загубитися кластери можуть в разі неправильного завершення роботи додатка або краху системи. Програми відновлення диска можуть виявити ці кластери й відновити їх.

У табл. 1.18 наведено приклад запису загублених кластерів у файловій структурі.

Таблиця 1.18

ЗАГУБЛЕНІ КЛАСТЕРИ У ФАЙЛОВІЙ СТРУКТУРІ

Каталог		
Ім'я	Початковий кластер	Розмір
(немає записів)	0	0
FAT 16		
Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	...
00999	0	Кластер доступний
01000	1001	Використається; посилання на наступний кластер
01001	1002	Використається; посилання на наступний кластер
01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	0	Кластер доступний
...	...	...
65526	0	Останній доступний кластер

З'являються ланцюжки кластерів, які не мають записів у каталозі. Найчастіше це відбувається в разі «зависання» програми під час відкриття файла.

Програми відновлення диска переглядають диск і створюють копію FAT в оперативній пам'яті. Потім ця копія порівнюється з «дійсною» FAT і в такий спосіб виявляються загублені кластери, тобто не приналежні жодному з наявних файлів. Практично всі програми відновлення можуть зберігати інформацію з загублених кластерів у файлі, а потім обнуляти їх.

Наприклад, програма *Chkdsk* з ланцюжків загублених кластерів створює файли з іменами FILE0001.CHK, FILE0002.CHK і т.д. Програма Chkdsk перетворить загублені кластери у файли так, як показано в табл. 1.19.

Таблиця 1.19

ЗАГУБЛЕНІ КЛАСТЕРИ ЗНАЙДЕНІ

Каталог		
Ім'я	Початковий кластер	Розмір
FILE0001.CHK	1000	4
FAT 16		
Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	...
00999	0	Кластер доступний
01000	1001	Використається; посилання на наступний кластер
01001	1002	Використається; посилання на наступний кластер
01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	0	Кластер доступний
...	...	...
65526	0	Останній доступний кластер

Як видно з наведеного прикладу, оригінальне ім'я файла не відновлюється. Однак його можна відновити, переглянувши вміст файлів, які створені програмою відновлення диска.

1.2.3.2. Пересічні файли

Такі файли з'являються, коли два записи каталогу неправильно вказують на один кластер. У результаті кластер «містить» дані з кількох файлів, що, природно, неприпустимо.

У табл. 1.20 наведено приклад запису файлової системи з пересічними файлами.

Таблиця 1.20

ПЕРЕСІЧНІ ФАЙЛИ

Каталог		
Ім'я	Початковий кластер	Розмір
Usconst.txt	1000	4
Pledge.txt	1002	2
FAT 16		
Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	...
00999	0	Кластер доступний
01000	1001	Використається; посилання на наступний кластер
01001	1002	Використається; посилання на наступний кластер
01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	0	Кластер доступний
...	...	...
65526	0	Останній доступний кластер

У розглянутому прикладі два файли займають два кластери — 1002 і 1003. Пересікання файлів починається з кластера 1002. Найчастіше один із пересічних файлів ушкоджений. Програми відновлення даних звичайно вирішують проблему пересічних файлів у такий спосіб: файли копіюються з новими іменами у вільне місце

диска, а пересічна область обох файлів (і їхні інші частини) видаляється. Зверніть увагу, що видаляються обидва файли, тобто усунення подібної помилки не породжує нових проблем: наприклад, запис у каталозі вказує на неіснуючий файл. Переглянувши два відновлені файли, можна визначити, який з них ушкоджений.

Для програм відновлення диска пошук пересічних файлів — дуже просте завдання, і практично всі дискові утиліти можуть вирішити дану проблему.

Хибний файл або каталог

Іноді інформація в записі каталогу для файла або підкаталогу не відповідає дійсності — запис містить кластер з неправильною датою або неправильним форматом. Практично всі програми відновлення диска усувають і цю проблему.

1.2.3.3. Помилки FAT

Як уже зазначалося, в разі ушкодження основної FAT доступ до файлів здійснюється за допомогою додаткової FAT. Програми відновлення диска повертають ушкоджену FAT в її справжнє місце розташування й активізують дзеркальне копіювання. Файлова система FAT 32 є більш спроможна до відновлення, оскільки в ній використовуються більш розвинені засоби дзеркального копіювання.

Приклад ушкодженої FAT наведено в табл. 1.21.

Таблиця 1.21

УШКОДЖЕНА FAT

Каталог		
Ім'я	Початковий кластер	Розмір
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	...
00999	0	Кластер доступний
01000	1001	Використається; посилання на наступний кластер
01001	0	Кластер доступний

Закінчення табл. 1.21

01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	0	Кластер доступний
...	...	...
65526	0	Останній доступний кластер

У розглянутому прикладі розмір файла в каталозі не відповідає кількості кластерів у FAT (загублено кластер 1001), а крім того, кластери 1002 і 1003 є загубленими. При відновленні даних ушкоджена FAT відновлюється з резервної копії. Практично всі програми відновлення даних успішно справляються з таким типом помилки FAT.

1.2.4. Відновлення диска й даних

Команди *Chkdsk, Recover* і *Scandisk* — «реанімаційна бригада» DOS, що займається відновленням ушкоджених даних на диску. Ці команди мають дуже простий і не надто дружній інтерфейс, їх застосування найчастіше впливає на систему, але іноді тільки вони й можуть допомогти. З перелічених утиліт найбільш відомі, мабуть, Recover, що відновлює програми, і *Chkdsk*, використовувана для перевірки файлової структури диска. Багато користувачів навіть не підозрюють, що *Chkdsk* може не тільки перевіряти, а й відновлювати ушкоджену файлову структуру диска. Ще одна програма — проста утиліта *Debug* — може допомогти вам у лиху, але тільки в тому разі, якщо ви точно знаєте, що і як робити.

Scandisk — утиліта, потужніша за *Chkdsk* і *Recover*; вона заміняє ці дві утиліти в DOS 6 і більш пізніх версіях, а також у Windows *9x*.

Опис команд *chkdsk* і *Recover* можна знайти в доповненні на компакт-диску, що додається.

1.2.4.1. Програма *Scandisk*

Програма *Scandisk* входить у поставку DOS версій 6 і більш пізніх, а також у Windows *9x*. Вона значно потужніша за утиліти *Chkdsk* і *Recover* і виконує функції їх обох. Програма *Scandisk* з Windows 95 OSR2 і Windows 98 може працювати з файловою системою FAT 32.

Програма *Scandisk* більше схожа на спрощену версію *Norton Disk Doctor* і дає змогу перевіряти як цілісність файлової структури, так і

роботу секторів на фізичному рівні. Виявивши помилки в каталогах або в FAT, *Scandisk* може їх виправити. Після визначення дефектного сектора в FAT позначається дефектний кластер, що містить цей сектор. При цьому програма намагається відновити ушкоджений файл, причому зберігаються дані як до дефектної ділянки, так і після неї.

У Windows *9x* є програма *Scandisk* для DOS і Windows. Файли цих програм називаються *Scandisk.exe* й *Scandiskw.exe* відповідно. Windows *9x* перевіряє диск у процесі встановлення операційної системи, а також після неправильного завершення роботи з системою. Ви можете також запустити програму *Scandisk* та її «віконну» версію з командного рядка.

Особливості роботи програми *Scandisk* ви можете знайти в книгах з операційних систем або в довідковій системі Windows *9x*.

1.2.4.2. Дефрагментація диска

Каталог		
Ім'я	Початковий кластер	Розмір
Pledge.txt	1002	2
Usconst.txt	1000	4

FAT 16

Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	...
00999	0	Кластер доступний

Як уже згадувалося вище, у файловій системі FAT дані в кластерах можуть розташовуватися в будь-якому місці диска. І при пошуку файла останній зчитується з кількох місць, що, природно, призводить до зниження продуктивності системи. Для переміщення файла в одне місце служать програми дефрагментації диска.

У Windows *9x* є програма дефрагментації диска, що працює з файловими системами FAT 16 і FAT 32. У Windows 98 у програму дефрагментації була додана функція прискорення запуску додатків — переміщення програм, що часто запускають, на початок диска.

Розглянемо роботу програми дефрагментації диска на прикладі. У табл. 1.22 наведено дані про розташування файлів у FAT.

Таблиця 1.22

ФРАГМЕНТОВАНІ ФАЙЛИ

Каталог		
Номер кластера	Значення	Призначення
01000	1001	Використається; посилання на наступний кластер
01001	1004	Використається; посилання на наступний кластер
01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	1005	Використається; посилання на наступний кластер
01005	FFFFh	Кінець файла
...	...	...
65526	0	Останній доступний кластер

У розглянутому прикладі файл Usconst.txt фрагментований на дві частини. Після запуску програми дефрагментації цей файл може бути розташований на диску так, як показано в табл. 1.23.

Таблиця 1.23

ФАЙЛ ДЕФРАГМЕНТОВАНИЙ

Каталог		
Ім'я	Початковий кластер	Розмір
Pledge.txt	1004	2
Usconst.txt	1000	4
FAT 16		
Номер кластера	Значення	Призначення
00002	0	Перший доступний кластер
...	...	...
00999	0	Кластер доступний
01000	1001	Використається; посилання на наступний кластер
01001	1002	Використається; посилання на наступний кластер

Закінчення табл. 1.23

01002	1003	Використається; посилання на наступний кластер
01003	FFFFh	Кінець файла
01004	1005	Використається; посилання на наступний кластер
01005	FFFFh	Кінець файла
...	...	...
65526	0	Останній доступний кластер

У процесі дефрагментації частини файла, розкидані по всьому диску, були з'єднані. Дефрагментація диска — досить тривалий процес, оскільки необхідно виконати велику кількість операцій читання й записування.

Окрім програми дефрагментації диска, що поставляється з операційною системою Windows *9x*, існують програми незалежних розробників. Прикладом однієї з таких програм може бути програма *Speed Disk* з комплекту *Norton Utilities*.

Слід пам'ятати, що процес дефрагментації — досить небезпечна процедура. Під час дефрагментації відбувається зчитування, видалення й перезапис даних. Збій у системі живлення при виконанні цієї операції може мати сумні наслідки. Тому перед дефрагментацією критично важливих даних треба виконувати резервне копіювання.

1.2.4.3. Програми незалежних розробників

Окрім стандартних програм для роботи з диском, що поставляються разом з операційною системою, існує величезна кількість дискових програм незалежних розробників. Найвідоміший пакет таких програм — *Norton Utilities* — розроблено фірмою *Symantec*. Більшість подібних програм створені для операційних систем DOS і Windows й можуть працювати з файловою системою FAT 32. Усі ці програми мають істотний недолік — їх треба діставати додатково, у той час як самі програми вже поставляються з операційною системою.

1.2.5. NTFS

Файлова система NTFS застосовується в операційній системі Windows NT. Незважаючи на те, що Windows NT може використо-

вувати розділи з FAT, файлова система NTFS забезпечує більші переваги порівняно з FAT: більші розміри файлів і розділів, додаткові атрибути файлів і розширені засоби безпеки. При розробці операційної системи Windows NT не існувало проблем забезпечення зворотної сумісності, тому файлова система має особливі властивості й підтримується тільки Windows NT.

Усі операційні системи Windows (крім Windows NT) засновані на DOS, тому всередині системи існує частина коду DOS. У Windows NT DOS-програми працюють у режимі емуляції DOS. При завантаженні іншої операційної системи розділ із NTFS недоступний.

У файловій системі ім'я файла може містити до 255 символів, включаючи пробіли, крапки й інші символи, крім * ? \ /; <> |. Оскільки NTFS — 64-розрядна файлова система, розмір файла й розділу може бути просто величезним — 2 Мбайт, або 17 179 869 184 Тбайт!

1.2.5.1. Архітектура NTFS

Незважаючи на існуючі розходження в структурі розділу файлових систем FAT і NTFS, вони мають подібні елементи, наприклад завантажувальну область. Розділ NTFS складається з **головної таблиці файлів** (*masterfile table — MFT*). MFT — це не те саме, що FAT. Замість використання таблиці з посиланнями на кластери, MFT містить більшу кількість інформації про файли і каталоги в розділі. У деяких випадках MFT може навіть мати файли і каталоги.

Перший запис у MFT називається **дескриптором** (*descriptor*) і містить інформацію про розташування самої MFT. Завантажувальний сектор у розділі NTFS має посилання на розташування запису дескриптора.

Другий запис в MFT — це дзеркальна копія дескриптора. Таке надлишкове зберігання даних забезпечує більшу стійкість до помилок.

Третій запис — це запис файла журналу. Всі операції (транзакції) в NTFS записуються в спеціальний файл журналу, що дає змогу відновити дані після збою. Інша частина MFT складається з записів для файлів і каталогів, які зберігаються в розділі. У файлі NTFS зберігаються атрибути, визначені користувачем і системою. Атрибути в розділі NTFS — це не прості прапорці з розділу FAT. Вся інформація про файл, тобто атрибути, у файловій системі NTFS зберігається разом із файлом і є частиною самого файла. Каталоги в NTFS складаються в основному з індексів файлів у цьому каталозі й не містять такої інформації про файл, як розмір, дата, час і ін.

Таким чином, MFT — це не просто список кластерів, а основна структура зберігання даних у розділі. Якщо файл або каталог відносно невеликий (до 1 500 байт), його запис може зберігатися в MFT. Для більших масивів даних MFT має покажчик файла або каталогу, а самі дані містяться в інших кластерах у розділі. Ці кластери називаються **екстентами** (*extents*). Усі записи в MFT, включаючи дескриптори й файл журналу, можуть використовувати екстенти для зберігання додаткових атрибутів. Атрибути файла, які є частиною запису MFT, називають **резидентними** (*resident*), а атрибути в екстентах, — **нерезидентними** (*nonresident*).

1.2.5.2. Сумісність NTFS

Дістати доступ до розділу NTFS з DOS і інших операційних систем не можна. Windows NT призначена для використання як мережна операційна система, тому доступ до файлів у розділі NTFS можна одержати за допомогою мережі. Для цього в NTFS підтримуються імена файлів, що задовольняють стандарт вісім-крапка-три.

Основна перевага файлової системи NTFS — це гарантування безпеки файлів і каталогів. Атрибути безпеки в NTFS називаються дозволами (permissions) і встановлюються системним адміністратором за допомогою надання доступу до даних на рівні прав користувачів і груп користувачів.

Однак ви можете встановити FAT-атрибути файлів у NTFS за допомогою стандартних інструментів операційної системи Windows NT, наприклад програми Windows NT *Explorer* або команди DOS *Attrib*. При копіюванні файлів з розділу NTFS у FAT усі атрибути файла зберігаються, і користувач із правами повного доступу не зможе видалити файл із FAT-атрибутом «тільки для читання».

Алгоритм створення коротких імен файлів у Windows NT практично такий самий, як і у файловій системі VFAT Windows 9x. Процес створення імені файла, що задовольняє стандарт вісім-крапка-три для операційних систем Windows *9x*, а також особливості цього процесу в Windows NT описані вище.

1.2.5.3. Створення розділу NTFS

Створити розділ NTFS можна тільки на жорсткому диску. Його не можна створити на дискеті, а на змінному пристрої, такому як *Iomega Zip* або *Jaz*, можна. Існує три способи створення розділу NTFS:

- при установці операційної системи Windows NT або після установки за допомогою програм роботи з диском;
- форматуванням існуючого розділу в NTFS (з видаленням усіх даних) за допомогою команди Format системи Windows NT (параметр */fs :ntfs*);

■ за допомогою перетворення існуючого розділу FAT в NTFS (зі збереженням усіх даних) під час або після встановлення Windows NT за допомогою програми *Convert*.

1.2.5.4. Інструменти для NTFS

У зв'язку з тим, що файлові системи NTFS і FAT різняться за своєю структурою, в NTFS не можна застосовувати дискові утиліти для FAT. Сама файлова система NTFS має засоби відновлення даних. Крім того, деякі дискові утиліти поставляються з операційною системою Windows NT. У NTFS необхідно використовувати програми дефрагментації диска, які випускаються незалежними виробниками, такі як програма *Diskeeper* фірми *Executive Software International, Inc*.

1.2.5.5. Найпоширеніші помилки та повідомлення про них

Тут ітиметься про найбільш поширені помилки файлових систем і способи їх усунення. Найчастіше з'являються такі системні повідомлення про помилки:

- *Missing Operating System*;
- NO ROM BASIC – SYSTEM HALTED;
- *Boot error Press Fl to retry*;
- *Invalid drive specification*;
- *Invalid Media Type*;
- *Hard Disk Controller Failule.*

Missing Operating System

Таке повідомлення про помилку вказує на проблеми в головному завантажувальному записі або записах таблиці розділу. Запис у таблиці розділу може вказувати на сектор, що не є початком розділу. Ця помилка можлива внаслідок розряду батареї на системній платі, що спричинює видалення параметрів BIOS.

Для розв'язання проблеми насамперед необхідно перевірити правильність установки параметрів у BIOS. Головний завантажувальний запис можна відновити за допомогою команди *Fdisk* /MBR. В інших випадках вирішити виниклу проблему можна за допомогою радикальних засобів — поділу диска на розділи та форматування

його, а відтак повторного встановлення операційної системи і необхідних додатків.

NO ROM BASIC — SYSTEM HALTED

Цю помилку генерує AMI BIOS у випадку ушкодження або відсутності завантажувального сектора або головного завантажувального запису на завантажувальному диску. Крім того, така помилка можлива в разі неправильного встановлення параметрів жорсткого диска в BIOS. Для вирішення цієї проблеми необхідно перевірити параметри диска в BIOS або ж відновити головний завантажувальний запис за допомогою команди *Fdisk* /MBR.

Boot error Press F1 to retry

Помилка генерується *Phoenix* BIOS у разі відсутності жорсткого диска або завантажувальних областей. Найбільш часта причина появи цієї помилки — відсутність активного розділу.

Invalid drive specification

Така помилка можлива тоді, коли жорсткий диск не поділено на розділи, записи таблиці розділів ушкоджені або ж містять хибні дані. Для усунення подібних проблем скористайтеся програмою Fdisk або програмами пакета *Norton Utilities*.

Invalid Media Type

Поява такого повідомлення свідчить, що, найімовірніше, ушкоджено (або не ініціалізовано) завантажувальний сектор, каталог або таблиця розміщення файлів. Наприклад, така помилка можлива, якщо диск поділено на розділи, але не відформатовано за допомогою команди *Format*.

Для усунення цієї помилки слід скористатися однією з програм відновлення диска або ж просто відформатувати його.

Hard Disk Controller Failule

Помилка такого типу з'являється внаслідок неправильно заданих параметрів накопичувача, установлених у BIOS, а також поганого підімкнення кабелів до накопичувача або системної плати. Для усунення цієї проблеми насамперед перевірте підімкнення накопичувача, а потім установіть в BIOS його правильні параметри.

1.2.6. Загальні способи розв'язання проблем із файловими системами

1.2.6.1. Вирішення проблем файлових систем у *MS DOS* та *Windows 9x/ME*

Для запобігання проблемам при доступі до жорсткого диска необхідно послідовно виконати такі дії:

1. Завантажте комп'ютер із завантажувальної дискети (її іноді називають аварійним диском). Це може бути завантажувальна дискета як DOS, так і Windows, головне, щоб на ній були записані такі програми: *Fdisk.exe*, *Format.com*, *Sys.com* і *Scandisk.exe*. Краще, якщо це буде аварійний диск з операційною системою Windows 95B.

2. Якщо із завантажувальної дискети не можна завантажити операційну систему, ймовірно, існують проблеми з апаратним забезпеченням.

3. Запустіть із завантажувальної дискети програму *Fdisk*. У меню виберіть вивід відомостей про наявні розділи (четвертий пункт меню).

4. Якщо відображається список розділів, перевірте наявність активного розділу — у стовпці стану біля одного з розділів має бути буква A.

5. Якщо в списку не відображається жодного розділу й ви не бажаєте відновлювати дані на диску, створіть новий розділ (або розділи), а потім відформатуйте його. При виконанні цих дій всі дані на диску будуть знищені.

6. Якщо вам необхідно відновити дані, скористайтесь однією з програм відновлення даних.

7. Якщо список розділів відображається й один із них активний, мабуть, ушкоджені системні файли. Для їхнього відновлення введіть команду Sys C:.

8. Тепер ваш жорсткий диск містить системні файли тієї операційної системи, що була на завантажувальному диску.

9. Витягніть дискету з дисководу й перезавантажте комп'ютер. Якщо й зараз при завантаженні з'являються помилки, вони скоріш за все викликані неправильною конфігурацією жорсткого диска в BIOS.

10. Запустіть програму *Scandisk* із завантажувального диска й перевірте диск на наявність помилок.

11. При перевірці диска за допомогою програми *Scandisk* не забудьте виконати перевірку поверхні диска. З появою великої

кількості ушкоджених секторів необхідно замінити накопичувач на жорстких дисках.

1.2.6.2. Вирішення проблем файлових систем у Windows 2000/XP

Способи розв'язання проблем файлової системи в операційних системах Windows 2000/XP практично ті самі, що й у Windows *9x*. Їх основна відмінність полягає у використанні службової програми *Recovery Console*, що входить у Windows 2000/XP.

Якщо програма *Recovery Console* введена в завантажувальне меню, запустіть систему у звичайному порядку, ввійдіть у систему як адміністратор, якщо це необхідно, і виберіть опцію *Recovery Console*.

У тому випадку, якщо програма Recovery Console не була попередньо введена в завантажувальне меню, завантажте систему за допомогою настановного компакт-диска Windows або диска Windows Setup. Виберіть опцію *Repair* у меню *Welcome to Setup* й натисніть клавішу <C> для запуску *Recovery Console*.

Якщо система не завантажується з настановного компакт-диска або завантажувальної дискети, то можливі деякі проблеми з апаратним забезпеченням. Перевірте тверді диски, конфігурацію BIOS і настановні параметри системної плати. Визначте гнучкий диск як перший завантажувальний пристрій, а CD-ROM — як другий, а потім запустіть знову систему.

Після запуску *Recovery Console* виконайте ряд таких дій:

1. Для одержання довідки і списку команд *Recovery Console* уведіть HELP у командному рядку.

2. Запустіть програму DISKPART, щоб одержати відомості про існуючий розділ диска.

3. Коли відобразиться перелік розділів, перевірте, чи визначено завантажувальний розділ як активний.

4. У тому випадку, якщо в списку не відображені наявні розділи й у вас немає бажання відновлювати які-небудь дані, що існують у даний час на диску, створіть новий розділ (або розділи) за допомогою команди FDISK. Для форматування створених розділів скористайтеся командою FORMAT. При виконанні цих дій усі дані на диску будуть знищені.

5. Якщо необхідно відновити дані, скористайтеся для цього однією з існуючих програм відновлення даних, наприклад *Norton Utilities* від компанії Symantec або *Lost and Found* від *Power Quest*.

6. Якщо при виконанні програми DISKPART відображається список розділів і один з них визначений як активний, виходить, можливе ушкодження системних файлів. Для їх відновлення введіть команду FIXBOOT.

7. Для того щоб запустити знову систему, уведіть команду EXIT. Перед цим витягніть завантажувальну дискету з дисководу A або ж настановний компакт-диск Windows 2000 з дисководу CD-ROM.

8. Якщо і після перезавантаження комп'ютера з'являться ті самі помилки, то вони, найімовірніше, викликані ушкодженням або неправильною конфігурацією твердого диска.

Запустіть знову *Recovery Console* і перевірте диск на наявність помилок, увівши команду CHKDSK у командному рядку.

Питання для самоперевірки

1. Що таке файлова система і як вона використовується?
2. Які структури, призначені для забезпечення доступу до файлів, підтримуються сучасними операційними системами?
3. Яке призначення завантажувального сектора логічного диска DOS?
4. Які дані містяться у таблиці розміщення файлів (FAT)?
5. У чому полягають особливості зберігання таблиці розміщення файлів (FAT)?
6. Що таке кластер і яке його призначення?
7. Як побудовано циліндр для діагностичних операцій читання й запису?
8. Яке призначення віртуальної таблиці розміщення файлів (VFAT)?
9. У чому полягає необхідність використання файлової системи з розширеними можливостями FAT32?
10. Чим відрізняється таблиця розміщення файлів FAT32 від попередніх версій файлових таблиць?
11. Які програми використовуються для відновлення диска й даних на дисках?
12. У чому полягають основні переваги файлової системи NTFS порівняно з FAT?
13. Які шляхи вирішення проблем файлових систем в *MS DOS*, *Windows 9x/ME* та *Windows 2000/XP*?

1.3. Діагностування технічних засобів комп'ютерних систем і мереж

1.3.1. Діагностика PC

Діагностичне програмне забезпечення необхідне в тому разі, якщо в системі починаються збої або якщо її модернізують додаванням нових пристроїв. Навіть для виконання простої операції (наприклад, установлення нової плати) чи пошуку несправності в апаратурі, що призвела до збою або «зависання» системи, необхідно мати досить повну інформацію про загальний стан комп'ютера. Завдяки діагностичним програмам можна перевірити роботу як усієї системи, так і окремих її вузлів.

Природно, що при експлуатації системи необхідне технічне обслуговування. Застосування регулярного обслуговування — запорука нормальної роботи комп'ютера і всієї комп'ютерної системи в цілому.

1.3.2. Діагностичні програми

Для PC існує кілька видів діагностичних програм (деякі з них поставляються разом із комп'ютером), що дають змогу користувачеві виявляти причини неполадок, що виникають у комп'ютері. У багатьох випадках такі програми можуть виконати основну роботу з визначення дефектного вузла. Умовно їх можна розділити на кілька груп, причому складність програм і їхні можливості в кожній наступній групі вище ніж у попередній.

До цього ряду можна віднести такі діагностичні програми:

- POST (*Power-On Self Test* — процедура самоперевірки при увімкненні) виконується при кожному ввімкненні комп'ютера.
- Діагностичні програми фірм-виробників. Більшість відомих фірм — виробників комп'ютерів (IBM, *Compaq*, Hewlett-Packard, *Dell* і т.д.) випускають для своїх систем спеціалізоване діагностичне програмне забезпечення, що звичайно містить набір тестів, які дають змогу ретельно перевірити усі компоненти комп'ютера.
- Діагностичні програми фірм — виробників устаткування. Багато виробників устаткування випускають діагностичні програми, призначені для перевірки певного пристрою. Наприклад, фірма *Adaptec* випускає програми для перевірки працездатності SCSI-адаптерів.

■ Діагностичні програми операційних систем. Операційні системи Windows, починаючи з найперших модифікацій, поставляються з кількома діагностичними програмами для перевірки різних компонентів комп'ютера.

■ Діагностичні програми загального призначення. Такі програми, що забезпечують ретельне тестування будь-яких PC-сумісних комп'ютерів, випускають багато фірм.

1.3.3. Самоперевірка при увімкненні (POST)

З початком випуску персональних комп'ютерів фірмами-виробниками було передбачено методи підвищення надійності, які раніше ніколи не застосовувалися в обчислювальній техніці. Мається на увазі програма POST і контроль парності пам'яті. Процедура POST розглядається як послідовність коротких підпрограм, що зберігаються в ROM BIOS на системній платі. Вони призначені для перевірки основних компонентів системи одразу після її увімкнення, що, власне, і є причиною затримки перед завантаженням операційної системи.

1.3.3.1. Що тестується

При кожному ввімкненні комп'ютера автоматично здійснюється перевірка його основних компонентів: процесора, мікросхем ROM, допоміжних елементів системної плати, оперативної пам'яті й основних периферійних пристроїв. Ці тести виконуються швидко і не дуже старанно порівняно з тестами, виконуваними діагностичними програмами. В разі виявлення несправного компонента видається попередження або повідомлення про помилку (несправність).

Хоча здійснювана програмою POST діагностика не зовсім повна, вона є першою програмою перевірки з числа таких програм, особливо якщо виявляються серйозні несправності в системній платі. Якщо виявиться, що неполадка досить серйозна, подальше завантаження системи буде припинено і з'явиться повідомлення про помилку, за яким, як правило, можна визначити причину несправності. Такі несправності іноді називають фатальними помилками (*fatal error*). Процедурою POST зазвичай передбачаються три способи індикації несправності: звукові сигнали, повідомлення, що виводяться на екран монітора, і шістнадцяткові коди помилок, які видаються в порт введення-виведення.

1.3.3.2. Звукові коди помилок, передбачені процедурою POST

У разі виявлення несправності процедурою POST комп'ютер видає характерні звукові сигнали, за якими можна визначити несправний елемент (або групу їх). Якщо комп'ютер справний, то при його увімкненні видається один короткий звуковий сигнал; якщо виявлено несправність, видається ряд коротких або довгих звукових сигналів, а іноді й комбінація їх. Характер звукових кодів залежить від версії BIOS і фірми — її розробника.

1.3.3.3. Повідомлення про помилки, видавані процедурою POST на екран комп'ютера

У більшості PC-сумісних моделей процедура POST відображає на екрані хід тестування оперативної пам'яті комп'ютера. Останнє виведене на екран число відповідає кількості пам'яті, що успішно пройшла перевірку. Так, може з'явитися повідомлення:

32768 KB OK

Загалом останнє виведене під час тестування число має збігатися з обсягом усієї встановленої в комп'ютері пам'яті (як основної, так і розширеної). Однак у деяких комп'ютерах може відображатися трохи менше значення, наприклад, у тому випадку, якщо не тестується уся верхня пам'ять UMA (*Upper Memory Area*) обсягом 384 Кбайт або її частина. Якщо по закінченні тестування число на екрані не відповідає загальному обсягові пам'яті, виходить, у системній пам'яті виявлена помилка.

Якщо під час виконання процедури POST виявлена несправність, на екран виводиться відповідне повідомлення, як правило, у вигляді числового коду з кількох цифр, наприклад:

1790-Disk 0 *Error*.

Визначити, якій несправності відповідає даний код, можна, скориставшись посібником з експлуатації і сервісного обслуговування для даного типу *PC*.

1.3.3.4. Коди помилок, видавані процедурою POST у порти введення-виведення

Менш відомою можливістю процедури *POST* є те, що на початку виконання кожного тесту за адресою спеціального порту введення-виведення програма видає коди тесту, що можуть бути прочитані

тільки за допомогою встановлюваної в рознімання розширення спеціальної плати адаптера. Такі адаптери зазвичай називаються POST-платами і розроблені для тестування системних плат задля виявлення можливих дефектів при їх виробництві (при цьому не потрібно підключати до них відеоадаптер і монітор). Зараз деякі фірми (*Micro* 2000, JDR Microde-vices, *Data Depot*, Ultra-X, *Quarterdeck*, *Trinitech* і ін.) випускають такі POST-плати для фахівців, що займаються сервісним обслуговуванням комп'ютерів (рис. 1.6).

POST-плата встановлюється в рознімання розширення. У момент виконання процедури POST на її вбудованому індикаторі будуть швидко мінятися двозначні шістнадцяткові числа. Якщо комп'ютер раптово припинить тестування або «зависне», у цьому індикаторі буде відображено код того тесту, під час виконання якого відбувся збій. Це дає змогу істотно звузити коло пошуку несправного елемента.

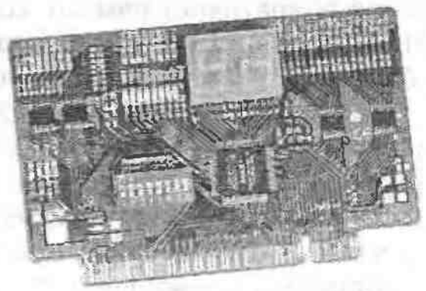


Рис. 1.6. Різновид POST-плати

Найчастіше використовуються тестові плати двох типів: шини, що вставляються в 8-розрядні частини рознімань, ISA або EISA і призначені для підключення до шини MCA. Деякі фірми роблять обидва види плат. Компанії *Micro* 2000 і *Data Depot* не випускають окрему MCA-плату — їхні універсальні пристрої дають можливість за допомогою додаткового адаптера підключати ISA/EISA-плату до MCA-шини. Інші фірми роблять лише ISA / EISA-плати й ігнорують шину MCA. На сьогодні більшість виробників випускають тестові плати тільки для шин PCI і ISA.

1.3.4. Діагностика окремих видів апаратного забезпечення

Багато типів діагностичного програмного забезпечення призначено для конкретних типів апаратного забезпечення. Ці програми поставляються разом із пристроями.

1.3.4.1. Діагностика SCSI-пристроїв

Більшість SCSI-адаптерів мають вбудовану BIOS, за допомогою якої можна настроювати адаптер і виконувати його діагностику. Наприклад, SCSI-адаптери, що випускаються фірмою *Adaptec*, поставляються з програмою SCSI *Select*, завдяки чому можна правильно зконфігурувати і протестувати працездатність адаптера.

1.3.4.2. Діагностика мережних адаптерів

Деякі виробники мережних плат, наприклад SMC і 3COM, також пропонують діагностичне програмне забезпечення. За допомогою цих програм можна перевірити інтерфейс шини, контроль пам'яті, встановленої на платі, вектори переривань, а також виконати циклічний тест. Ці програми можна знайти на дискеті або компакт-диску, що поставляється разом із пристроєм, або ж звернутися на *Web*-вузол виробника.

1.3.4.3. Діагностичні програми загального призначення

Існує безліч різноманітних діагностичних програм для PC-сумісних комп'ютерів. Є спеціальні програми для тестування пам'яті, жорстких дисків, дисководів гнучких дисків, відеоадаптерів й інших компонентів системи. Одні з них займають гідне місце серед такого роду програм, інші явно не дотягають до професійного рівня. Програми, орієнтовані на користувачів із середньою підготовкою, виконані не дуже ретельно, і позбавлені багатьох можливостей, необхідних для професійної роботи. У цьому розділі йтиметься про деякі діагностичні програми.

Більшість тестових програм можна запускати в пакетному режимі, що дає змогу без втручання оператора виконати ряд тестів. Можна скласти програму автоматизованої діагностики, найбільш ефективну в тому випадку, якщо вам необхідно виявити можливі дефекти або виконати однакову послідовність тестів на кількох комп'ютерах. Ці програми перевіряють усі типи системної пам'яті: основну (*base*), розширену (*expanded*) і додаткову (*extended*). Місце несправності найчастіше можна визначити з точністю до окремої мікросхеми або модуля (SIMM або DIMM).

1.3.5. Діагностичні програми операційної системи

У більшості випадків придбавати діагностичну програму недоцільно, оскільки систему можна протестувати наявними засобами операційної системи. До складу Windows практично всіх версій починаючи з найперших входять кілька діагностичних програм.

1.3.5.1. MSD (*Microsoft Diagnostics*)

Починаючи з DOS 6JC I Windows 9X, фірма *Microsoft* стала включати до складу цих систем мало кому відому програму MSD (*Microsoft Diagnostics*). Насправді це скоріше програма для конфігурації системи, ніж повноцінна програма діагностики. Вона дає змогу досить швидко вирішити проблеми загального використання переривань і розподілу пам'яті.

MSD повідомляє основну інформацію про версії BIOS, тип процесора, відеоадаптер, мережу (якщо вона є), мишу, дисководи, CD-ROM, паралельні та послідовні порти і версії DOS. Крім того, ви можете довідатися про завантажені в пам'ять драйвери пристроїв і резидентні програми (це придасться при розв'язанні конфлікту між двома програмами й особливо в разі спроби розмістити одразу кілька програм у пам'яті). MSD може в графічній формі показати їх розташування в пам'яті — це більш наочно, ніж текст, видаваний командою MEM із комплекту постачання операційної системи DOS.

MSD входила у комплект постачання Windows 9X. Вона не копіюється на жорсткий диск при установці операційної системи, але міститься на компакт-диску з Windows 9X. Для запуску цієї програми необхідно перезавантажити комп'ютер з Windows 9X у режим MS DOS, а потім запускати цю діагностичну програму.

1.3.5.2. Диспетчер пристроїв

Диспетчер пристроїв у Windows версій 2000, *XP* і вище є більш досконалим ніж у попередніх. Він знаходиться в розділі Система (*System*) вікна Панель управления (*Control Panel*) на вкладці Оборудование (*Device Manager*) (рис. 1.7). У цій вкладці відображається встановлене в комп'ютері устаткування. Тут ви можете зконфігурувати кожен пристрій, переглядати займані ним ресурси й обновляти драйвери.

У перемикачі Вид можна задавати відтворення пристроїв та ресурсів за типом і підключенням. Встановивши перемикач Вид у положення Устройства по подключению (*View devices by connection*), можна переглянути різні порти й інтерфейси комп'ютера.

Якщо відкрити піктограму Компьютер (*Computer*), відкриється його тип (наприклад, Стандартный компьютер), двічі клацнувши на цьому елементі, можна отримати діалогове вікно Свойства: Стандартный компьютер, у якому можна переглянути інформацію про розподіл переривань, про драйвери, що використовуються, а також, у разі необхідності, запустити майстер діагностики (рис. 1.8).

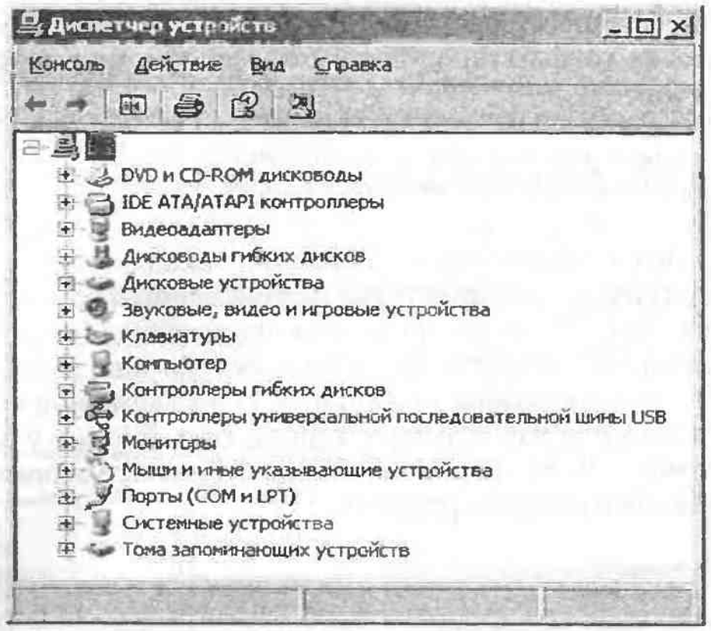


Рис. 1.7. Диспетчер пристроїв Windows *XP*

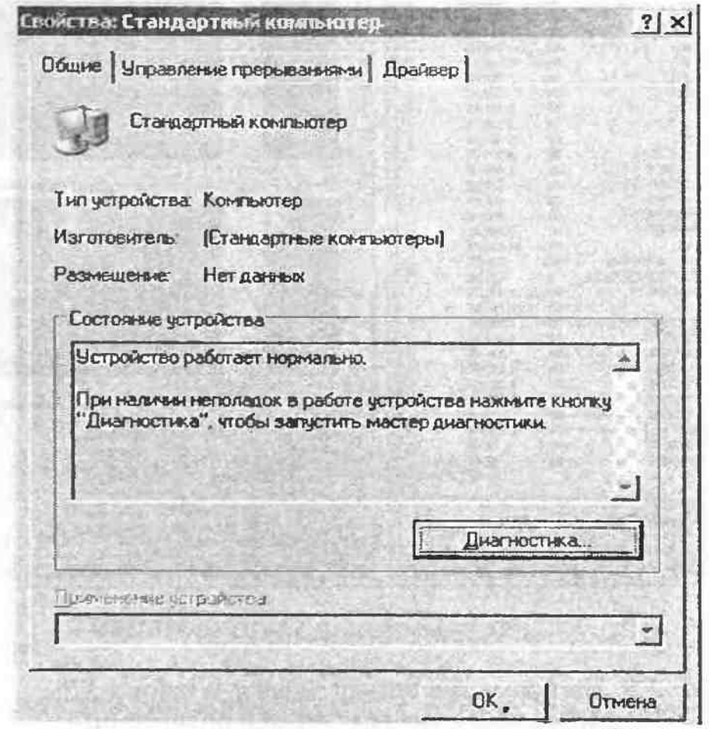
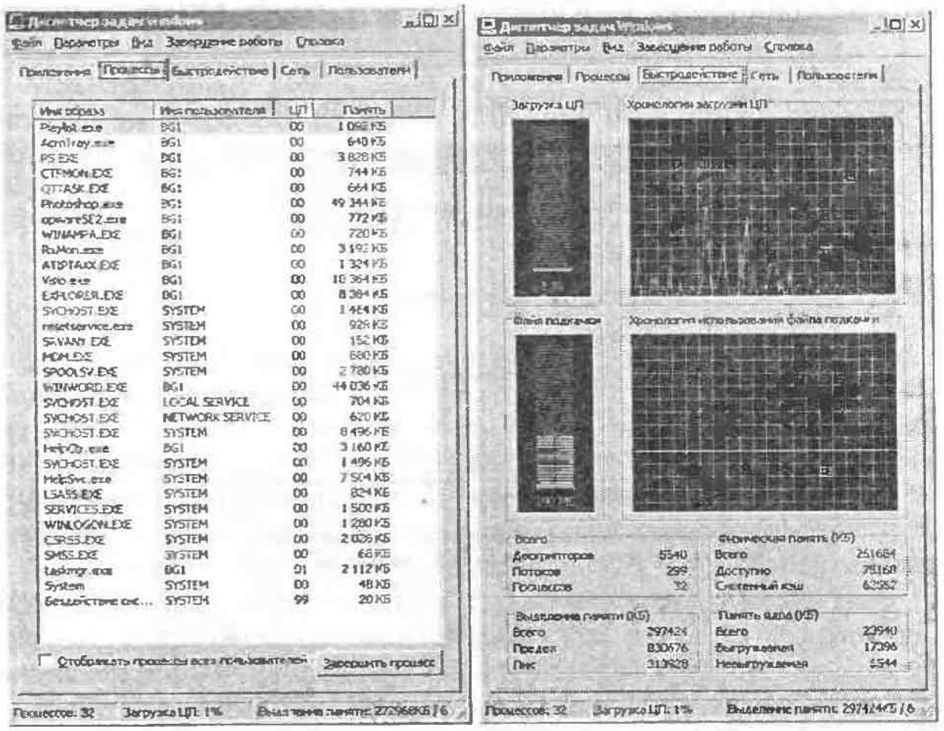


Рис. 1.8. Діалогове вікно Свойства: Стандартный компьютер

Якщо при установці пристроїв *Plug and Play* у Windows *XP* між ними виникають конфлікти, за допомогою диспетчера пристроїв їх можна усунути.

1.3.6. Диспетчер задач

За допомогою Диспетчера задач, що входить до складу і Windows *XP*, дуже просто визначити параметри системи, наприклад використання пам'яті і її розподіл між процесами, які відображаються у вигляді таблиці (рис. 1.9, *а*), параметри швидкодії, такі як Загрузка ЦП і роботи файлів підкачки, відображаються в графічному вигляді, а дані про пам'ять ядра, кеш-пам'ять й ін. — у спеціальних вікнах (рис. 1.9, *б*). В разі необхідності можна отримати дані про використання мережних ресурсів.



а *б*

Рис. 1.9. Програма Диспетчер задач у Windows *XP*:
а — розподіл ресурсів пам'яті між процесами;
б — дані про швидкодію та розподіл фізичної пам'яті й пам'яті ядра

1.3.6.1. Загальні діагностичні програми

У системі Windows *XP* програму **«Сведения о системе»** включено до складу службових у розділі **«Стандартные»** (рис. 1.10). Ця програма просто незамінна при дослідженні комп'ютера, оскільки з її допомогою можна одержати детальну інформацію про ресурси апаратури, склад і параметри компонентів, програмне середовище і т.д.

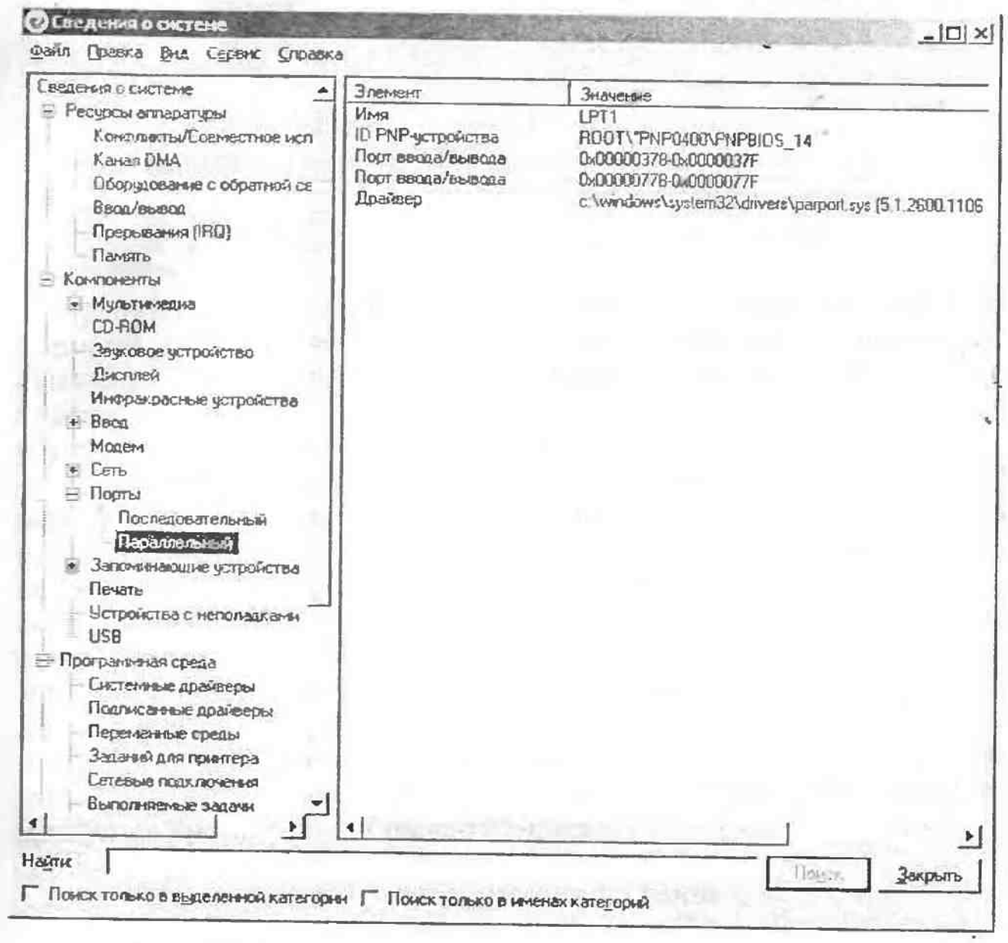


Рис 1.10. Програма **«Сведения о системе»** у Windows *XP*

Питання для самоперевірки

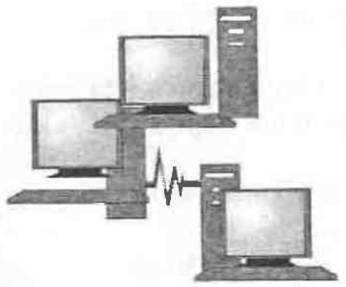
1. Які програми належать до складу діагностичних програм ПК?
2. Які види тестування проводяться при включенні ПК?

3. Як відображається на екрані ПК хід тестування процедурою POST?

4. Які програми входять до складу сучасних операційних систем для діагностування технічних засобів ПК?

5. У чому полягає призначення диспетчера пристроїв операційної системи *Windows XP*?

6. Які параметри системи визначаються за допомогою задач, що входить до складу *Windows XP*?



Модуль 2

Процеси експлуатаційного обслуговування комп'ютерних систем і мереж

2.1. Адміністрування користувачів з використанням локальних і глобальних груп

2.1.1. Користувачі, ресурси й операції доступу

Адміністрування користувачів полягає у створенні облікової інформації користувачів (визначальне ім'я користувача, приналежність користувача до різних груп користувачів, пароль користувача), а також у визначенні прав доступу користувача до ресурсів мережі — комп'ютерів, каталогів, файлів, принтерів тощо.

Створення облікової інформації користувачів здійснюється в мережі *Windows* NT утилітою *User Manager* для локального комп'ютера й *User Manager for Domains* — для всіх комп'ютерів домену. Права доступу до ресурсів задаються в мережі *Windows* NT різними засобами, залежно від типу ресурсу. Можливість використання комп'ютерів *Windows* NT *Workstation* як робочої станції — за допомогою *User Manager for Domains*, доступ до локальних каталогів і файлів (тільки для файлової системи NTFS, що підтримує права доступу) — за допомогою засобів *Windows* NT *Explorer*, до вилучених розподілюваних (*share*) каталогів — за допомогою *Server Manager*, доступ до принтерів — з панелі *Printers*.

2.1.1.1. Типи користувачів

У мережі *Windows* NT можуть бути визначені такі типи користувачів і груп користувачів:

✓ локальний інтерактивний користувач комп'ютера (користувач локальної облікової бази даних, працює з ресурсами комп'ютера інтерактивно);

✓ локальний мережний користувач комп'ютера (користувач локальної облікової бази даних, працює з ресурсами комп'ютера через мережу);

✓ користувач домену (користувач глобальної облікової бази даних домену на PDC);

✓ локальна група комп'ютера (може створюватися на всіх комп'ютерах домену, крім PDC і BDC, у яких вона вироджується в локальну групу домену);

✓ локальна група домену — складається з користувачів домену (тільки на PDC);

✓ глобальна група домену — складається з користувачів домену (може входити в локальну групу домену).

Для кожного типу груп є деякий набір вбудованих груп: *Administrators*, *Server Operators*, *Users*, *Everyone*, *Domain Users* і ін.

Для однозначної ідентифікації глобальної групи в багатодоменній мережі використовується складене її ім'я, наприклад *Marketing\Managers*, де *Marketing* — ім'я домену, *Managers* — ім'я глобальної групи.

2.1.1.2. Типи об'єктів

Каталоги й файли. Процедури завдання правил доступу розрізняються для локальних і розподілюваних каталогів і файлів. Операції: *read, full control, change, add, ...*

Принтери.

Операційна система. Стосовно цього типу об'єктів визначаються права з виконання різних сервісів й утиліт: вхід, архівування файлів, зміна конфігурації панелей *Program Manager*, ...

2.1.1.3. Типи операцій доступу

Операції доступу — це дії об'єктів над суб'єктами. Операції можуть бути або дозволені, або заборонені, або взагалі не мати сенсу для даної пари об'єкта й суб'єкта.

Уся безліч операцій поділяється на підмножини, що мають особливі назви:

■ дозволу (*permissions*) — це операції, які можуть бути визначені для суб'єктів усіх типів стосовно об'єктів типу файл, каталог або принтер;

■ права (*user rights*) — визначаються для об'єктів типу група на виконання деяких системних операцій: створення резервних копій, вимикання комп'ютера (*shutdown*) і т.п. Права призначаються за допомогою *User Manager for Domains*;

■ можливості користувачів (*user abilities*) — визначаються для окремих користувачів на виконання дій, пов'язаних із формуванням їхнього операційного середовища, наприклад, зміна складу програм-

них груп, показуваних на екрані дисплея, включення нових іконок в *Desktop*, можливість використання команди *Run* тощо.

Права й дозволи, дані групі, автоматично надаються її членам, даючи змогу адміністраторові розглядати велику кількість користувачів як одиницю облікової інформації.

Можливості користувачів визначаються профілем користувача.

2.1.2. Локальні, глобальні й спеціальні групи користувачів

Windows NT *Server* використовує три типи груп користувачів: локальні, глобальні й спеціальні. Кожний тип має своє призначення, можливості й обмеження.

Локальна група може визначатися для домену або для комп'ютера. Локальні групи дають користувачам права й дозволи на ресурси того комп'ютера (або домену), де зберігається облікова інформація локальної групи. Доступ до ресурсів комп'ютера — *Windows* NT *Workstation* або *Windows* NT *Server* — може бути визначений тільки для членів локальної групи цього комп'ютера, навіть якщо ці комп'ютери є членами домену. Наприклад, доступ до ресурсів сервера *Windows* NT *Server* 2 на рис. 2.1 може бути визначений тільки для користувачів, облікові дані яких зберігаються в SAM 2 цих комп'ютерів.

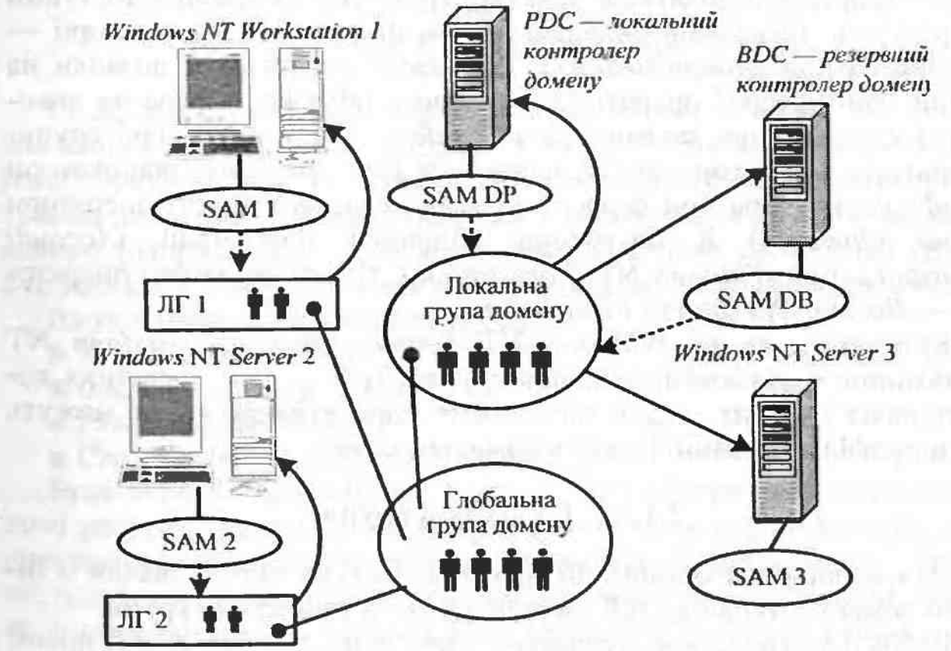


Рис. 2.1. Приклад локальної групи

Оскільки база SAM PDC копіюється на всі BDC домену, то користувачі, певні в PDC, можуть мати права на ресурси як PDC, так і всі BDC домену.

Доступ до ресурсів комп'ютера для користувачів домену забезпечується завдяки механізму включення в локальну групу окремих користувачів домену та його глобальних груп. Включені користувачі й групи дістають ті самі права доступу, що й решта членів даної групи. Механізм включення глобальних груп у локальні є основним засобом централізованого адміністрування прав доступу в домені *Windows* NT.

2.1.2.1. Локальні групи

Локальна група не може містити інші локальні групи. Тому в мережі, що використовує модель робочої групи, немає можливості визначити на одному комп'ютері всіх користувачів мережі й надавати їм доступ до ресурсів інших комп'ютерів.

У будь-якому разі локальна група поєднує деяку кількість користувачів і глобальних груп, яким надається спільне ім'я — ім'я локальної групи. Локальні групи можуть включати користувачів і глобальної групи не тільки даного домену, а й будь-яких доменів, що довіряють їм.

Windows NT *Workstation* і *Server* підтримують кілька вбудованих локальних груп для виконання системних завдань. Адміністратор може створювати додаткові локальні групи для керування доступом до ресурсів. Вбудовані локальні групи діляться на дві категорії — адміністратори (*Administrators*), які мають усі права й дозволи на даний комп'ютер, і оператори, які мають обмежені права на виконання специфічних завдань. Для *Windows* NT *Server* є такі групи-оператори: оператори архівування (*Backup Operator*), реплікатори (*Replicator*), оператори сервера (*Server Operator*), принт-оператори (*Print Operator*) й оператори облікової інформації (*Account Operator*). Для *Windows* NT *Workstation* є тільки дві групи операторів — *Backup Operators* і *Power Users*.

Крім того, як на *Windows* NT *Server*, так і на *Windows* NT *Workstation* є вбудовані локальні групи *Users* — для звичайних користувачів і *Guests* — для тимчасових користувачів, які не можуть мати профілю й повинні мати мінімальні права.

2.1.2.2. Глобальні групи

Для спрощення організації надання доступу користувачам з іншого домену в *Windows* NT уведене поняття глобальної групи.

Глобальна група користувачів — це група, що має ім'я й права, глобальні для всієї мережі, на відміну від локальних груп користу-

вачів, які мають імена й права, дійсні тільки в межах одного домену. Адміністратор, що довіряє домену, може надавати доступ до ресурсів свого домену користувачам із глобальних груп тих доменів, яким даний домен довіряє. Глобальні групи можна включати до складу локальних груп користувачів ресурсного домену.

Глобальна група — це деяка кількість користувачів одного домену, які групуються під спільним ім'ям. Глобальним групам можуть даватися права й дозволи через включення їх у локальні групи, які вже мають необхідні права й дозволи. Глобальна група може містити тільки облікову інформацію користувачів з локальних облікових баз даних, вона не може мати локальні групи або інші глобальні групи.

Існує три типи вбудованих глобальних груп: адміністратор домену (*Domain Admins*), користувачі домену (*Domain Users*) і гості домену (*Domain Guests*). Ці групи завжди були і є членами локальних груп адміністраторів, користувачів і гостей відповідно.

Вбудовані групи слід використовувати всюди, де тільки це можливо. Рекомендується формувати їх у такій послідовності:

- ✓ в обліковому домені необхідно створити користувачів і добавити їх до глобальних груп;
- ✓ включити глобальні групи до складу локальних груп ресурсних доменів;
- ✓ надати локальним групам необхідні права й дозволи.

2.1.2.3. Спеціальні групи

Спеціальна група — використовується тільки *Windows* NT *Server* для системного доступу. Спеціальні групи не містять облікової інформації користувачів і груп. Адміністратори не можуть приписати користувачів до цих груп. Користувачі або є членами цих груп за замовчуванням (наприклад, кожний користувач є членом спеціальної групи *Everyone*), або вони стають ними залежно від своєї мережної активності.

Існує чотири типи спеціальних груп:

- *Network* (Мережева).
- *Interactive* (Інтерактивна).
- *Everyone* (Кожний).
- *Creator Owner* (Творець-Власник).

Будь-який користувач, що хоче дістати доступ до розподілюваного ресурсу мережі, автоматично стає членом групи *Network*. Користувач, який локально ввійшов у комп'ютер, автоматично включається в групу *Interactive*. Той самий користувач залежно від того, як він працює з комп'ютером, матиме різні права. Будь-який користувач мережі є членом групи *Everyone*. Адміністратор може призна-

чити групі *Everyone* будь-які права. При цьому адміністратор може надати будь-які права користувачеві, не заводячи на нього облікової інформації на своєму комп'ютері. Група *Creator Owner* містить облікову інформацію користувача, що створив ресурс або володіє ним.

У файловій системі NTFS дозволи групі *Creator Owner* даються на рівні каталогу. Власник будь-якого каталогу або файла, створеного в даному каталозі, дістає дозволи, дані групі *Creator Owner*. Наприклад, можна призначити якому-небудь каталогу для членів групи *Everyone* дозвіл *Read* (Читання), а групі *Creator Owner* надати доступ *Full Control* (Повне керування). Будь-який користувач, що створює файли або підкаталоги в цьому каталозі, буде мати до них доступ *Full Control*.

2.1.2.4. Вбудовані групи користувачів і їхні права

Права визначаються для об'єктів типу групи на виконання деяких системних операцій: створення резервних копій, вимикання комп'ютера (*shutdown*) і т.п. Права призначаються за допомогою *User Manager for Domains* (табл. 2.1).

Таблиця 2.1

ПРАВА ДЛЯ ВБУДОВАНИХ ЛОКАЛЬНИХ ГРУП ДОМЕНУ СТОСОВНО СИСТЕМИ, ЩО ВИКОНУЄ РОЛЬ PDC АБО BDC

Права та вбудовані права	Administrators	Server Operators	Account Operators[1]	Print Operators	Backup Operators	Everyone[2]	Users[3]	Guests
Права								
Log on locally (локальний логічний вхід)	*	*	*	*	*	O	O	O
Access this computer from network (доступ до даного комп'ютера через мережу)	*	O	O	O	O	*	O	O
Take ownership of files (установлення прав власності на файли)	*	O	O	O	O	O	O	O
Manage auditing and security log (керування аудитом і обліком подій, пов'язаних з безпекою)	*	O	O	O	O	O	O	O
Change the system time (зміна системного часу)	*	*	O	O	O	O	O	O
Shutdown the system (зупинка системи)	*	*	O	O	*	O	O	O
Force shutdown from remote system (ініціація зупинки з вилученої системи)	*	*	O	O	O	O	O	O

Продовження табл. 2.1

Права та вбудовані права	Administrators	Server Operators	Account Operators	Print Operators	Backup Operators	Everyone	Users	Guests
Backup files and directories (резервне копіювання файлів і каталогів)	*	*	*	*	*	O	O	O
Restore files and directories (відновлення файлів і каталогів зі стрімера)	*	*	O	O	*	O	O	O
Load and unload device drivers (завантаження й вивантаження драйверів пристроїв)	*	O	O	O	O	O	O	O
Add workstation to domain (додавання робочих станцій до домену)	*	O	O	O	O	O	O	O
Убудовані права								
Create and manage user accounts (створення й керування користувацькою обліковою інформацією)	*	O	*1	O	O	O	O	O
Create and manage global groups (створення й керування глобальними групами)	*	O	*1	O	O	O	O	O
Create and manage local groups (створення й керування локальними групами)	*	O	*1	O	O	O	*2	O
Assign user rights (призначення прав для користувачів)	*	O	O	O	O	O	O	O
Manage auditing of system events (керування аудитом системних подій)	*	O	O	O	O	O	O	O
Lock the server (блокування сервера)	*	*	O	O	O	*3	O	O
Override the lock of the server (подолання блокування сервера)	*	*	O	O	O	O	O	O
Format server's hard disk (форматування жорсткого диска сервера)	*	*	O	O	O	O	O	O
Create common groups (створення спільних груп)	*	*	O	O	O	O	O	O
Keep local profile (зберігання локального профілю)	*	*	*	*	*	O	O	O
Share and stop sharing directories (розподіл і припинення розподілу каталогів)	*	*	O	O	O	O	O	O

Закінчення табл. 2.1

Права та вбудовані права	Administrators	Server Operators	Account Operators	Print Operators	Backup Operators	Everyone	Users	Guests
Share and stop sharing printers (розподіл і припинення розподілу принтерів)	*	*	O	*	O	O	O	O

[1]Оператори облікової інформації (*Accounts Operators*) не можуть змінювати облікову інформацію адміністраторів або ж змінювати глобальну групу *Domain Admins*, або локальні групи *Administrators*, *Server Operators*, *Account Operators*, *Print Operators*, або *Backup Operators*.

[2]Хоча *Everyone* має право блокувати сервер, тільки користувачі, які можуть також входити локально в цей сервер, можуть насправді його заблокувати.

[3]Хоча члени групи *Users* мають право створювати локальні групи домену, але вони не зможуть ним скористатись, якщо їм не дозволено входити локально в сервер або не дозволено користуватись утилітою *User Manager for Domains*.

Схожі права можна задати й стосовно *Windows* NT *Server*, не виконуючу роль PDC або BDC — за допомогою утиліти *User Manager for Domains*, а також до *Windows* NT *Workstation* — за допомогою утиліти *User Manager*.

2.1.3. Можливості користувачів

Можливості користувачів визначаються для окремих користувачів на виконання нечисленних дій, що стосуються реорганізації їхнього операційного середовища:

- ✓ включення нових програмних одиниць (іконок) у групу програм панелі *Program Manager*;
- ✓ створення програмних груп *Program Manager*;
- ✓ надання імен складу програмних груп;
- ✓ зміна властивостей програмних одиниць (наприклад, включення в стартову групу);
- ✓ запуск програм із меню FILE у *Program Manager*;
- ✓ установлення з'єднань із мережевим принтером (крім тих, які вже передбачені в профілі користувача).

Можливості користувача є частиною так званого профілю користувача (*User Profile*), який можна змінювати за допомогою утиліти *User Profile Editor*. Профіль поряд з описаними можливостями включає й встановлення середовища користувача на його робочому комп'ютері: кольори, шрифти, набір програмних груп і їх склад.

2.1.3.1. Дозвіл на доступ до каталогів і файлів

Адміністратор може управляти доступом користувачів до каталогів і файлів у розділах диска, відформатованих під файлову сис-

тему NTFS. Розділи, відформатовані під FAT і HPFS, не підтримуються засобами захисту *Windows* NT. Однак можна захистити розподілювані по мережі каталоги незалежно від того, яка використовується файлова система.

Для захисту файла або каталогу необхідно встановити для нього дозвіл (*permissions*). Кожний установлений дозвіл визначає вид доступу, що користувач або група користувачів мають стосовно даного каталогу або файла. Наприклад, коли ви встановлюєте дозвіл *Read* до файла MY IDEAS.DOC для групи COWORKERS, користувачі з цієї групи можуть переглядати дані цього файла і його атрибути, але не можуть змінювати файл або видаляти його.

Windows NT дає змогу використати набір стандартних дозволів, які можна встановлювати для каталогів і файлів. Стандартними дозволами для каталогів є: *No Access, Read, Add, Add&Read, Change* і *Full Control*.

Стандартними дозволами для файлів є:

No Access, *Read*, *Change* і *Full Control*.

Стандартні дозволи являють собою групи індивідуальних дозволів. Кожному стандартному дозволові відповідає певна установка фіксованого набору індивідуальних дозволів. Індивідуальні дозволи можуть бути:

Read (R), *Write* (W), *Execute* (X), *Delete* (D),

Change Permission (P), *Take Ownership* (O).

При установці стандартного дозволу поруч із ним у дужках відображаються заголовні букви встановлених індивідуальних дозволів. Наприклад, при установці для файла стандартного дозволу *Read* поруч зі словом *Read* з'являється абревіатура RX, яка означає, що стандартному дозволу *Read* відповідає установка двох індивідуальних дозволів — *Read* і *Execute*.

Адміністратор може за допомогою утиліти *File Manager* установлювати як стандартні, так і індивідуальні дозволи.

Для того, щоб ефективно користуватися можливостями механізмів безпеки NTFS, потрібно пам'ятати таке:

користувачі не можуть користуватися каталогом або файлом, якщо вони не мають дозволи на це або ж вони не належать до групи, що має відповідний дозвіл.

Дозволи мають накопичувальний ефект за винятком дозволу *No Access*, що скасовує всі інші наявні дозволи. Наприклад, якщо група CO-WORKERS має дозвіл *Change* для якогось файла, а група *Finance* має для цього файла тільки дозвіл *Read* і Петров є членом обох груп, то в Петрова буде дозвіл *Change*. Однак якщо дозвіл для групи *Finance* зміниться на *No Access*, то Петров не зможе використати цей файл, незважаючи на те, що він — член групи, що має доступ до файла.

Коли ви створюєте в каталозі файли й підкаталоги, то вони успадковують дозволи, які має каталог.

Користувач, що створює файл або каталог, є власником (*owner*) цього файла або каталогу. Власник завжди має повний доступ до файла або каталогу, бо може змінювати дозволи для нього. Користувачі — члени групи *Administrators* — можуть завжди стати власниками будь-якого файла або каталогу.

Найзручнішим шляхом керування захистом файлів і каталогів є установка дозволів для груп користувачів, а не для окремих користувачів. Звичайно користувачеві потрібен доступ до багатьох файлів. Якщо користувач — член якої-небудь групи, що має доступ до цих файлів, то адміністраторові простіше позбавити користувача цих прав, видаливши його зі складу групи, а не змінювати дозволи для кожного файла. Зазначимо, що установка дозволу для індивідуального користувача не скасовує дозволів, даних користувачеві як члену деякої групи.

Для каталогу індивідуальні дозволи мають такий сенс (табл. 2.2):

Таблиця 2.2

Функції	R	W	X	D	P	O	FC
Переглядати імена файлів у каталозі	*	O	*	O	*	O	*
Переглядати атрибути каталогу	*	O	*	O	*	O	*
Додавати файли й підкаталоги	O	*	O	*	O	*	*
Змінювати атрибути каталогу	O	*	O	*	O	*	*
Переходити в підкаталоги каталогу	O	*	*	O	*	O	*
Переглядати власника каталогу й дозволу	*	I	*	O	*	O	*
Видаляти каталог	O	*	O	*	O	*	*
Змінювати дозволи каталогу	O	*	O	*	*	O	*
Ставати власником каталогу	O	*	O	*	O	*	O

Для файла індивідуальні дозволи мають такий сенс (табл. 2.3):

Таблиця 2.3

Функції	R	W	X	D	P	O	FC
Переглядати дані файла	*	O	O	O	O	O	*
Переглядати атрибути файла	*	O	*	O	O	O	*
Змінювати атрибути файла	O	*	O	O	O	O	*
Змінювати й додавати дані у файл	O	*	O	O	O	O	*

Закінчення табл. 2.3

Функції	R	W	X	D	P	O	FC
Виконувати файл, якщо це програма	O	O	*	O	O	O	*
Переглядати власника файла й дозволу	*	*	*	O	O	O	*
Видаляти файл	O	O	O	*	O	O	*
Змінювати дозволи файла	O	O	O	O	*	O	*
Ставати власником файла	O	O	O	O	O	*	*

Для файлів є така відповідність індивідуальних і стандартних дозволів файла:

- ✓ *No Access* Жодного
- ✓ *Read* RX
- ✓ *Change* RWXD
- ✓ *Full Control* Всі дозволи

Стандартні дозволи для каталогу являють собою об'єднання індивідуальних дозволів для каталогу й для файлів, що входять у цей каталог:

✓ *No Access*	(Жодного)	(Жодного)
✓ *List*	(RX)	(Не визначені)
✓ *Read*	(RX)	(RX)
✓ *Add*	(WX)	(Не визначені)
✓ *Add&Read*	(RWX)	(RX)
✓ *Change*	(RWXD)	(RWXD)
✓ *Full Control*	(Всі дозволи)	(Всі дозволи)

2.1.3.2. Керування профілями користувачів

Коли користувач локально входить перший раз у який-небудь комп'ютер, то для нього за замовчуванням створюється профіль. Всі настроювання середовища (кольори тла, шпалери, шрифти тощо) автоматично зберігаються в підкаталозі *Profiles* системного каталогу даного комп'ютера, наприклад, C:\NT40w\Profiles\username, де *username* — ім'я користувача. Профіль зберігається у файлі з ім'ям *ntuser.dat*

Адміністратор також може налаштовувати профіль користувача, входячи в який-небудь комп'ютер під ім'ям цього користувача.

На відміну від профілю користувача, що встановлюється за замовчуванням, існує також *Roaming* — переміщуваний профіль користувача, що формує те саме середовище для даного користувача, незалежно від того, з якого комп'ютера він увійшов у мережу.

Профілі користувачів, що переміщуються, зберігаються централізовано на сервері, а не на локальних комп'ютерах користувачів.

Адміністратор може визначити для користувача один із двох типів переміщуваних профілів.

1. Індивідуальний переміщуваний профіль, який користувач може змінювати. Будь-які зміни, внесені користувачем у своє середовище, вносяться в індивідуальний переміщуваний профіль тоді, коли користувач логічно виходить із мережі. Коли той самий користувач входить знову, з сервера завантажується останній варіант профілю. Таким чином, якщо використовуються переміщувані індивідуальні профілі, то в кожного користувача є свій власний переміщуваний профіль. Цей профіль зберігається у файлі *ntuser.dat* в одному з розподілюваних каталогів сервера.

2. Обов'язковий *(mandatory)* переміщуваний профіль — це заздалегідь сконфігурований адміністратором профіль, який користувач не може змінити. Один обов'язковий профіль може бути призначено кільком користувачам. Цей вид профілю доцільно призначати тим користувачам, яким потрібне однакове середовище, наприклад, операціоністам банку. Обов`язковий профіль повинен мати розширення *.man*. Індивідуальний профіль можна зробити обов'язковим, перейменувавши його з *Ntuser.dat* у *Ntuser.man*.

Починаючи з версії 4.0 адміністраторові пропонується потужніший засіб керування профілями користувачів — *System Policy Editor*. З його допомогою адміністратор може змінювати профіль користувача, не входячи під його ім'ям. При цьому він може встановлювати обмеження, які неможливо було б установити, входячи під ім'ям користувача, наприклад, заборона на застосування команди *Run*. *System Policy Editor* може використатися для формування як локальних, так і переміщуваних профілів. Переміщуваний профіль зберігається у файлі *Ntconfig.pol* у розподілюваному каталозі *Netlogon* на PDC.

2.1.4. Аудит

2.1.4.1. Призначення аудита

Аудит — це функція *Windows* NT, що дає змогу відстежувати діяльність користувачів, а також всі системні події в мережі. За допомогою аудита адміністратор одержує інформацію:

- ✓ про виконану дію;
- ✓ про користувача, що виконав цю дію;
- ✓ про дату й час виконання дії.

Адміністратор використовує політику аудита (Audit Policy) для вибору типів подій, які потрібно відстежувати. Коли подія відбувається, у журнал безпеки того комп'ютера, на якому вона відбулася, додається новий запис. Журнал безпеки є тим засобом, за допомогою якого адміністратор відстежує настання тих типів подій, які він задав.

Політика аудита контролера домену визначає кількість і тип зафіксованих подій, що відбуваються на всіх контролерах домену. На комп'ютерах *Windows* NT *Workstation* або *Windows* NT *Server*, що входять у домен, політика аудита визначає кількість і тип зафіксованих подій, що відбуваються тільки на даному комп'ютері.

Адміністратор може встановити політику аудита для домену для того, щоб:

- ■ відстежувати успішні й неуспішні події, такі як логічні входи користувачів, читання файлів, зміни в дозволах користувачів і груп, виконання мережних з'єднань тощо;
- ■ виключити або мінімізувати ризик неавторизованого використання ресурсів;
- ■ аналізувати часові тенденції, використовуючи архів журналу безпеки.

Аудит становить частину системи безпеки. Коли всі засоби безпеки відмовляють, записи в журналі виявляються єдиним джерелом інформації, на підставі якої адміністратор може зробити висновки про те, що відбулося або готується відбутися в системі.

Установлення політики аудита є привілейованою дією: користувач повинен або бути членом групи *Administrators* на тому комп'ютері, для якого встановлюється політика, або мати права *Manage auditing and security log*.

2.1.4.2. Реалізація політики аудита

Політика аудита встановлюється окремо для кожного комп'ютера. Наприклад, для аудита логічного входу користувачів у домен необхідно встановити політику аудита на PDC (ця сама політика визначена й для всіх BDC домену). Для спостереження за доступом до файлів на сервері домену — *member* server — необхідно встановити політику аудита на цьому сервері.

Події записуються в журнал певного комп'ютера, але можуть переглядатися з будь-якого комп'ютера мережі користувачем, що має права адміністратора на той комп'ютер, де відбулася подія.

Установка політики аудита включає два етапи:

- ✓ визначення політики аудита за допомогою панелі *Audit Policy* утиліти *User Manager for Domains* або *User Manager*;
- ✓ визначення каталогів, файлів і принтерів, доступ до яких необхідно відстежувати (для цього використовується *Windows* NT

Explorer або панель *Printers*, спостереження за файлами й каталогами можливо тільки для файлової системи NTFS).

Перегляд журналу подій здійснюється за допомогою утиліти *Event Viewer* (журнал *Security*).

2.1.5. Реплікація каталогів у мережі Windows NT

Іноді в мережі корисно мати кілька копій того самого файла на різних комп'ютерах. Прикладами таких файлів може бути файл із номерами телефонів працівників підприємства, інші довідкові дані, потрібні одночасно багатьом клієнтам мережі. Тому для зниження навантаження на сервер, що зберігає такий файл, доцільно розмістити копії (репліки) цього файла на кількох серверах мережі й розподілити навантаження на ці сервери *між* клієнтами мережі.

Для підтримки синхронізму між даними різних копій файла застосовується схема майстра-копії файла. Одна копія файла є майстром-копією, тобто оригіналом, у якому дозволяється робити зміни. Інші версії цього файла створюються копіюванням по мережі майстра-копії. Процес копіювання майстра-копії на сервери мережі називається реплікацією. Звичайно реплікація виконується або періодично, або в разі виникнення змін у майстра-копії.

Сервіс реплікації *Windows* NT дає змогу автоматично реплікувати файли, що перебувають у певному каталозі будь-якого комп'ютера, у каталоги інших комп'ютерів мережі.

Іншим прикладом необхідності реплікації файлів, характерним для мережі *Windows* NT, є необхідність реплікації файлів вхідних сценаріїв (*Logon Scripts*) користувачів і файлів системної політики (*System Policy Files*) у тому випадку, коли в мережі, крім основного, наявний резервний контролер домену.

Це пов'язане з тим, що коли користувач автентифікується на контролері домену (на основному або резервному), то контролер за замовчуванням шукає файл вхідного сценарію або файл системної політики у своєму певному локальному каталозі. Наприклад, файл системної політики повинен за замовчуванням перебувати в каталозі system-root\system32\Repl\Import\Scripts (що має розподілюване ім'я NETLOGON). Тому для того, щоб користувач мав той самий профіль, заданий у файлі системної політики, незалежно від того, який контролер домену його автентифікує, потрібно помістити копії цього файла на всі контролери домену.

Модель реплікації *Windows* NT включає такі елементи:

■ сервер-експортер. Цей сервер реплікує обновлювані файли з певного призначеного для реплікації каталогу на комп'ютер;

■ сервер-імпортер. Таким сервером може бути тільки комп'ютер під керуванням *Windows* NT *Server*.

Комп'ютер-імпортер приймає репліковані файли від комп'ютера-експортера. Комп'ютер-імпортер, може приймати обновлені файли від певного комп'ютера-експортера, або ж від усіх комп'ютерів-експортерів домену. Як комп'ютери-імпортери можуть виступати комп'ютери під керуванням *Windows* NT *Server*, *Windows* NT *Workstation*, LAN *Manager for* OS/2 *Server*.

Каталоги експорту й імпорту. Сервер експортує файли з підкаталогів головного каталогу експорту. За замовчуванням цим каталогом є каталог systemroot\System32\Repl\Export. Для експортування файлів адміністратор повинен створити в цьому каталозі підкаталоги для кожної групи файлів, що мають експортуватися. Слід зазначити, що файли, поміщені безпосередньо в каталог systemroot\System32\Repl\Export, експортуватися не будуть. Наприклад, для експорту файлів *Logon Scripts* необхідно помістити їх у підкаталог system-root\System32\Repl\Export\Scripts.

Кожний комп'ютер-імпортер повинен мати головний каталог імпорту, що відповідає головному каталогові експорту. За замовчуванням це каталог system-root\System32\Repl\Import.

Сервіс реплікації каталогів Directory Replicator service. Цей сервіс керує реплікацією файлів. Він працює на кожному сервері-експортері та комп'ютері-імпортері. Сервіс реплікації копіює з головного каталогу комп'ютера-експортера всі підкаталоги разом із файлами в головний каталог імпорту кожного комп'ютера-імпортера. При зміні якого-небудь файла в підкаталогах головного каталогу експорту сервіс реплікації копіює його у відповідний підкаталог імпорту.

Сервіс реплікації на кожному комп'ютері, що бере участь у процесі реплікації, повинен працювати під ім'ям деякого вигаданого користувача, наприклад, *repl*., спеціально створеного для цих цілей. Конфігурування сервісу для входу під ім'ям користувача виконується з панелі *Services*. Даний користувач має бути членом груп *Backup Operator* і *Replicator* для копіювання файлом реплікації в обхід можливих заборон із прав доступу.

Сервіс реплікації періодично перевіряє стан експортованих файлів для виявлення змін. Коли зміна виявлена, то відбуваються такі дії:

• сервер-експортер посилає повідомлення про зміну комп'ютерам-імпортерам або по домену;

• коли комп'ютер-імпортер одержує повідомлення, він звертається до сервера експорту й читає структуру каталогу експорту;

• комп'ютер-імпортер копіює всі нові або змінені файли у свої підкаталоги імпорту, а також видаляє зі своїх підкаталогів імпорту ті файли, які відсутні в підкаталогах експорту.

Параметри сервісу реплікації перебувають в *Registry*:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Serveces\Replicator\Parameters.

2.1.6. Приклади практичного використання методів адміністрування

2.1.6.1. Створення й видалення користувачів

Вправа зі створення користувачів на комп'ютері з російськомовною версією Windows NT Workstation

Створіть ряд нових локальних користувачів.

Увага! Щоб уникнути проблем із запам'ятовуванням паролів, варто призначати новим користувачам паролі, що збігаються з їхніми іменами.

Для створення облікових записів нових користувачів потрібно використати утиліту *Диспетчер пользователей*.

Перейменувати користувача *Администратор* в *adm*_ws1 (або *adm*_ws2, *adm*_ws3, ... — відповідно до імені комп'ютера), для того щоб можна було в подальших експериментах не плутати адміністраторів різних доменів і працівників станцій один з одним.

Створіть для груп *Администратор*, *Пользователи* й *Опытные пользователи* нових користувачів, дайте їм імена з приставками типу _ws1, наприклад, *Nik*_ws2.

Для користувачів груп *Пользователи* й *Опытные пользователи* перевірити їхні можливості з використання ресурсів домену.

Вправа зі створення користувачів на комп'ютері з англомовною версією Windows NT Server

Користувачі, що працюють за серверами *Windows* NT *Server*, входять у них як у домен, тобто, наприклад, в DOM1 і т.п., і створюють нових користувачів домену.

Для створення облікових записів нових користувачів потрібно використати утиліту *User Manager for Domains*.

Користувач *Administrator* перейменовується в *adm*_dom1 (*adm*_dom2, ...) — відповідно до імені домену, для того щоб можна було в подальших експериментах не плутати адміністраторів різних доменів і працівників станцій один з одним.

Задайте для груп *Domain Admins*, *Domain Users* і *Server Operators* нових користувачів з іменами з приставками _dom1 і т.п.

Для нових користувачів домену (крім адміністраторів) виконайте такі дії:

✓ задайте інтервал часу, коли користувачеві дозволений вхід у систему — пункт *Hours* у меню *New User*;

✓ забороніть вхід з деяких комп'ютерів у межах домену (для цього в панелі *New User* потрібно вибрати *Log on To* й у вікні діалогу вибрати *May Log On To These Workstation*, у повому вікні діалогу, що з'явилося, задайте імена тих комп'ютерів, з яких буде дозволений вхід);

✓ встановіть дату спливання терміну дії облікової інформації даного користувача — пункт *Account* у меню *New User*;

✓ перевірте дію уведених установок, виконуючи логічні входи в домен;

✓ переконайтеся в можливості транзитної автентифікації — вході в домен з будь-якого комп'ютера домену.

2.1.6.2. Дослідження дозволів (*permissions*)

Створіть новий каталог у розділі C: локального диска, помістіть в нього кілька файлів з наявних каталогів шляхом копіювання

Перевірте, які індивідуальні дозволи стосовно цього каталогу встановлені операційною системою за замовчуванням. Для цього використайте пункт *Properties* меню *File* папки, у яку входить новий каталог. У закладці *Security* натисніть кнопку *Permissions*.

Задайте каталогу стандартний дозвіл CHANGE для одного з нових користувачів, наприклад, *user*_dom1. Для цього в закладці *Security* натисніть кнопку *Permissions*, а потім кнопку *Add* (перед цим корисно видалити дозвіл *Full Control* для групи *Everyone*).

Відзначте в наданому списку, які дії ви можете виконати стосовно цього каталогу:

- переглядати імена файлів у каталозі;
- переглядати атрибути каталогу;
- додавати файли й підкаталоги;
- змінювати атрибути каталогу;
- переходити в підкаталоги каталогу;
- переглядати власника каталогу й дозволу;
- видаляти каталог;
- змінювати дозволи каталогу;
- ставати власником каталогу.

Виберіть один із файлів даного каталогу. Перевірте, які дії ви можете виконати стосовно цього файла. Заповніть таблицю:

✓ переглядати дані файла;

✓ переглядати атрибути файла;
✓ змінювати атрибути файла;
✓ змінювати й додавати дані у файл;
✓ виконувати файл, якщо це програма;
✓ переглядати власника файла й дозволу;
✓ видаляти файл;
✓ змінювати дозволи файла;
✓ ставати власником файла.

Після перевірки прав доступу до локальних файлів:

- зробіть новий каталог розподілюваним (*share*) і задайте для нього право *Read*;
- увійдіть по мережі в робочу станцію домену під тим же ім'ям, що й у вправі зі створення нового каталогу у розділі *C* (тобто, *user_dom*1);
- перевірте, які права є стосовно цього каталогу при доступі по мережі.

2.1.6.3. Дослідження прав користувачів (*User Rights*)

Мета дослідження: з'ясувати різницю між правами користувачів комп'ютера (*Windows NT Workstation і Windows NT Server*), правами користувачів домену, правами користувачів локальної групи комп'ютера, локальної групи домену і глобальної групи.

Локальний користувач комп'ютера може:

✓ інтерактивно увійти у свій комп'ютер, якщо він має право *Log on locally* і якщо комп'ютер не заблокований іншим користувачем (блокування переборюється тільки адміністратором даного комп'ютера);

✓ увійти в комп'ютер по мережі, якщо він є локальним користувачем цього комп'ютера й наділений правом *Access this computer from network*;

✓ користуватися ресурсами свого комп'ютера відповідно до прав і дозволів, наданих йому за замовчуванням або адміністратором;

✓ доступ до ресурсів інших серверів домену (з огляду на те, що він не користувач домену) йому дозволений тільки тоді, коли він є й локальним користувачем кожного сервера.

Користувач домену (не є локальним користувачем комп'ютера, з якого він входить) може користуватися ресурсами всіх серверів домену з правами вбудованого користувача *User*, якщо він включений у вбудовану глобальну групу *Domain Users* (за замовчуванням глобальна група *Domain Users* включається в локальну групу *Users* всіх серверів домену).

Адміністратор домену (входить у глобальну групу *Domain Admins*) може:

- заводити користувачів і групи (локальні й глобальні) домену на PDC;
- приєднувати комп'ютери до домену;
- переглядати розподілювані ресурси (*share*) на серверах домену (крім *Windows for Workgroups*), створювати й міняти дозволи на доступ до них;
- створювати локальних користувачів на комп'ютерах *Windows NT Workstation*.

2.1.6.4. Керування профілями користувачів

Для створення переміщуваних профілів користувача:

✓ відкрийте папку «Система» панелі керування;

✓ виберіть закладку *User Profiles*, скопіюйте профіль адміністратора в яку-небудь ПОРОЖНЮ папку, у діалозі копіювання змініть права використання профілю;

✓ зробіть папку, що містить скопійований профіль, розподілюваною;

✓ у диспетчері користувачів укажіть шлях до папки профілю в графі профілю користувача;

✓ якщо профіль повинен бути примусовим, перейменуйте в папці профілю ntuser.dat в ntuser.man.

Для установки обов'язкових профілів (системних політик) у домені за допомогою *System Policy Editor* необхідно:

- створити нову політику за допомогою *System Policy Editor*, установити в ній необхідні параметри й обмеження середовища користувача за допомогою іконки *Default User*;
- зберегти файл політики під ім'ям *ntconfig.pol* у поділюваному каталозі NETLOGON на контролері вашого домену (поділюваний каталог NETLOGON відповідає каталогу NT40*s**System*32*Repl**Import**Scripts*);
- увійти в робочу станцію домену під будь-яким ім'ям і перевірити, чи діють установки, зроблені у файлі політики, на середовище користувача.

2.1.7. Засоби перегляду мережевих ресурсів

2.1.7.1. Типи сервісів перегляду

Сервіс перегляду комп'ютерів (*Windows* NT *Computer Browser*) призначено для створення списку доменів і серверів, наявних у ме-

режі. Користувач переглядає цей список і вибирає потрібний ресурс. Цей сервіс працює, коли користувач робить запит на перегляд мережі з командного рядка або натискає кнопку CONNECT NETWORK DRIVE програми *File Manager*.

Сервіси перегляду можуть працювати як на *Windows* NT *Workstation*, так і на *Windows* NT *Server*. Є чотири типи сервісів перегляду.

Головний сервіс перегляду домену (*Domain Master Browser*). Цей тип сервісу завжди працює на первинному контролері домену. Він відповідає за збір інформації для всього домену, включаючи всі підмережі, і передає список ресурсів домену резервним сервісам перегляду домену (*Backup Browsers*).

Резервний сервіс перегляду (*Backup Browser*). Цей сервіс працює з копією списку перегляду й поширює його по запитах серед комп'ютерів домену. Всі комп'ютери *Windows* NT Server автоматично конфігуруються з установленим резервним сервісом перегляду. Комп'ютери *Windows* NT *Workstation* мають потенційну можливість виконувати резервний сервіс перегляду, якщо в мережі менше трьох комп'ютерів *Windows* NT *Server* виконують функції головного й резервного перегляду.

Головний сервіс перегляду (*Master Browser*). Цей тип сервісу одержує інформацію від комп'ютерів робочої групи або домену й передає список ресурсів домену резервному сервісові перегляду. В інтермережі з глобальними зв'язками в кожної підмережі виконується свій головний сервіс перегляду, а в локальній мережі цю функцію виконує головний сервіс перегляду домену. Комп'ютер для розміщення головного сервісу перегляду вибирається автоматично відповідно до деякої евристичної процедури. У порядку зменшення значимості враховується тип операційної системи (наприклад, *Windows* NT переважніше, ніж *Windows for Workgroups*), період знаходження у ввімкненому стані (чим довше вже пропрацював комп'ютер, тим його шанси вище), ім'я комп'ютера (лексикографічне впорядкування — сервер АС має перевагу перед сервером АУ).

Головний сервіс перегляду (*Preferred Master Browser*) — це сервер, що спеціально сконфігурований для перемоги в процедурі виборів головного сервісу переглядів. Це особливо важливо в мережі TCP/IP для виконання функції перегляду ресурсів домену й дозволу імен *Net*BIOS, якщо в мережі не реалізована служба WINS.

Для того, щоб комп'ютер став виконувати кращий головний сервіс перегляду, треба встановити параметр Is Domain Master Browser у стан *True*, або *Yes* (цей параметр за замовчуванням установлюється в стан *False*, або *No*, навіть якщо комп'ютер є в цей момент головним «переглядачем»). Параметр перебуває в базі даних Registry.

Потенційний «переглядач» — це будь-який комп'ютер, що може бути обраний як головний або резервний «переглядач». Комп'ютер *Windows* NT *Server* завжди є головним або резервним «переглядачем», а комп'ютери *Windows* NT *Workstation* і *Windows for Workgroups* мають потенційну можливість стати головними або резервними «переглядачами».

2.1.7.2. Перегляд глобальної мережі

У мережах *Windows* NT кожна локальна мережа є незалежною одиницею перегляду, для якої повинен бути передбачений власний головний і резервний сервіси перегляду. Тому вибори головного сервісу перегляду обмежені межами локальної мережі.

Підтримка перегляду глобальної мережі реалізована тільки в *Windows* NT *Server*. Тому в кожній локальній мережі, що входить у глобальну корпоративну мережу, має бути принаймні один комп'ютер *Windows* NT *Server*.

Комп'ютери з головними сервісами перегляду домену відповідають за створення списку наявних комп'ютерів у доменах і підмережах глобальної мережі.

Якщо домен включає небагато локальних підмереж, з'єднаних глобальними зв'язками (рис. 2.2), то головний «переглядач» кожної підмережі використає спрямовані дейтаграми під назвою «Оголошення головного переглядача» (*Master Browser Announcement*) для повідомлення про себе «Головному переглядачеві домену». Для створення списку серверів домену «головний переглядач» домену посилає запити всім «головним переглядачам», які оголосили про себе.

Потім «головний переглядач» домену поєднує власний список серверів зі списками серверів «головних переглядачів». Цей процес повторюється кожні 15 хвилин і гарантує, що «головний переглядач» домену буде мати повний список всіх серверів домену. Коли клієнт виконує запит про перегляд серверів, такий як *net view*, до «головного переглядача», то головний переглядач зможе надати повний список серверів домену, а не тільки локальної підмережі.

Робоча група *Windows* NT не може покривати невелику кількість підмереж. Будь-яка робоча група *Windows* NT, що покриває частку підмережі, буде працювати як кілька окремих робочих груп. Це також стосується робочих груп *Windows for Workgroups*.

Якщо «головний переглядач» не є первинним контролером домену, то він повинен синхронізувати свою інформацію з «головним переглядачем» домену, що працює в складі первинного контролера

домену. Ця синхронізація виконується кожні 15 хвилин через відправлення запиту «головному переглядачеві» домену.

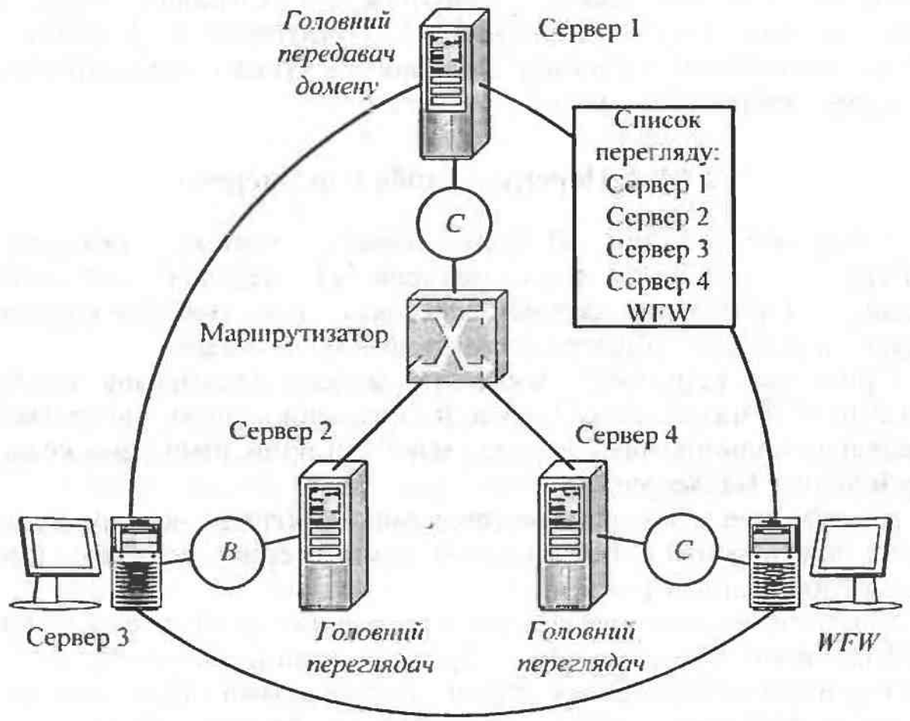


Рис. 2.2. Перегляд мережних ресурсів у глобальній мережі

Крім цього, головний переглядач відсилає спрямовані дейтаграми «Оголошення головного переглядача» («*Master Browser Announcement*») «головному переглядачеві» домену. Цей додатковий рівень синхронізації гарантує, що «головний переглядач» матиме повний список всіх серверів домену, незалежно від того, до якої підмережі ці сервери належать.

2.1.7.3. Перегляд кількох доменів

У мережі, що складається з кількох доменів, клієнти мають потребу в засобах перегляду списків:

— доменів;

— серверів у конкретному домені.

Windows NT надає такі можливості для своїх клієнтів. *Windows* NT додає новий інформаційний рівень для функції перегляду мережі (net view), що дає змогу клієнтам одержувати список доступних доменів від «головного переглядача».

Після того, як комп'ютер стає «головним переглядачем», він починає широкомовно передавати повідомлення «Оголошення домену» («*Domain Announcement*») щохвилини протягом перших 5 хвилин його роботи як «головного переглядача». Після сплівання цього строку оголошення поширюються кожні 15 хвилин.

Якщо домен не повідомляє про себе протягом трьох зазначених інтервалів, його ім'я має бути вилучене зі списку доменів. Через це домен може з'являтися в списках перегляду ще протягом 45 хвилин після його зупинки.

«Головний переглядач» одержує пакети «Оголошення домену» від інших доменів і поміщає їхні імена у свій список перегляду ресурсів.

«Головний переглядач» може ініціювати процес оголошення доменів про себе. Головний переглядач використає цю можливість тільки тоді, коли його список імен доменів порожній, наприклад, у випадку, коли потенційний переглядач стає «головним переглядачем».

Пакет «Оголошення Домену» містить:

— Ім'я домену.

— Ім'я головного переглядача для цього домену.

— Інформацію про те, яка операційна система встановлена на комп'ютері-переглядачі — *Windows* NT *Workstation* або *Windows* NT *Server*.

Якщо на комп'ютері-переглядачі працює *Windows* NT *Server*, то в пакеті також вказується, чи є комп'ютер і первинним контролером домену.

У разі, коли «головний переглядач» домену відмовляє, «головні переглядачі» бачитимуть тільки сервери, які перебувають у їхніх локальних мережах. Це означає, що сервери, котрі перебувають поза локальною мережею, будуть послідовно вилучені зі списку перегляду.

Оскільки головний переглядач домену одночасно є й первинним контролером домену, адміністратор може усунути наслідки відмови через переклад резервного контролера домену в режим первинного контролера домену. Вторинний контролер домену виконує багато функцій первинного контролера, але не може автоматично стати первинним контролером у разі відмови первинного.

2.1.7.4. Перегляд інтермережі TCP/IP

Сьогодні «переглядачі» взаємодіють в основному за допомогою широкомовних розсилань.

У середовищі глобальних мереж (таких як мережі TCP/IP, але не IPX), у яких домени розділяються маршрутизаторами, із широкомовним трафіком можуть виникнути проблеми, оскільки за замовчуванням широкомовні пакети не передаються маршрутизаторами. Розглянемо два питання:

Як контролери доменів, розділені маршрутизаторами, виконують функції перегляду?

Як локальні клієнти переглядають ресурси вилучених доменів, які не належать до їхньої локальної мережі?

Файл LMHOSTS. Дозвіл імен у протоколі *Net*BIOS на сьогодні виконується за допомогою широкомовних пакетів, і цей механізм працює тільки в межах локальної підмережі. Для дозволу імен комп'ютерів, підключених до інших підмереж, необхідно сконфігурувати файл LMHOSTS. Цей файл повинен містити відображення імен на IP-адреси всіх комп'ютерів, підключених не до даної локальної підмережі.

Для забезпечення взаємодії між підмережами й головним «переглядачем» домену адміністратор повинен сконфігурувати файл LMHOSTS так, аби він містив імена *Net*-BIOS і IP-адреси всіх «переглядачів».

Для того, щоб «головний переглядач» кожної підмережі дістав доступ до первинного контролера домену, інформація про первинний контролер домену має втримуватися у файлі LMHOSTS кожного «головного переглядача» (із приміткою #DOM). Ці самі вимоги ставляться й до резервних контролерів домену.

Файл LMHOSTS «головного переглядача» кожної підмережі повинен містити таку інформацію:

- ✓ IP-адреса й *Net*BIOS-ім'я кращого «головного переглядача».
- ✓ Ім'я домену, якому передують префікс #PRE і #DOM.

Наприклад:

130.20.7.80 Ім'я_кращого_головного_переглядача
#PRE #DOM: ім'я_домену.

Використання порту 137 протоколу UDP (*Net*BIOS *Name Service Broadcast*). У глобальних мережах не завжди виникають проблеми з широкомовним трафіком. Деякі маршрутизатори можуть бути сконфігуровані так, що вони мають змогу поширювати певні типи широкомовних пакетів, у той час як інші фільтруються.

Усі широкомовні пакети протоколу NetBIOS зверху TCP/IP (NBT) посилають на порт 137 протоколу UPD, що визначений як сервіс імен NBT. Це призначення порту визначається стандартами RFC 1001 і 1002. Звичайно такі пакети фільтруються маршрутизаторами, так вони надсилаються із широкомовними адресами MAC і IP

рівнів. Однак деякі маршрутизатори просувають пакети, відправлені із зазначеним номером порту UDP.

У результаті «переглядач» поводиться так, ніби він перебував в одній великій підмережі. Всі домени й комп'ютери будуть у цьому випадку побачені всіма системами, включаючи й комп'ютери *Windows for Workgroups*.

Сервіс WINS. При реалізації в мережі сервісу WINS непотрібно конфігурувати файли LMHOSTS або використовувати в маршрутизаторах порту 137 протокол UDP.

Використання служби WINS вимагає виконання таких умов:

Служба WINS конфігурується на комп'ютері, що працює під керуванням *Windows* NT *Server* 3.5.

Комп'ютери-клієнти підтримують сервіс WINS. Сервіс WINS підтримують комп'ютери під керуванням *Windows* NT 3.5, *Windows for Workgroups* (зі стеком TCP/IP-32), а також комп'ютери із клієнтською оболонкою *Microsoft Network Client*, що поставляє на компакт-диску *Windows* NT *Server* 3.5.

Клієнти, що не підтримують службу WINS, повинні, як і раніше, мати сконфігурований файл LMHOSTS для перегляду мережі із глобальними зв'язками, навіть якщо сервіс WINS реалізований у домені.

Питання для самоперевірки

1. Які типи користувачів і груп користувачів визначені у мережі *Windows* NT?
2. На які категорії діляться вбудовані локальні групи для виконання системних завдань, що підтримують *Windows* NT *Workstation* і *Server*?
3. Які можливості визначаються для окремих користувачів при реорганізації їхнього операційного середовища?
4. Призначення функції аудиту у *Windows* NT.
5. Яку інформацію одержує адміністратор за допомогою функції аудиту у мережі *Windows* NT?
6. Як отримується дозвіл на доступ до каталогів і файлів у мережі *Windows* NT?
7. Як здійснюється керування профілями користувачів у мережі *Windows* NT?
8. Як здійснюється реплікація каталогів у мережі *Windows* NT?

2.2. Організація технічного обслуговування комп'ютерних систем і мереж

2.2.1. Основні експлуатаційно-технічні показники комп'ютерних систем

2.2.1.1. Загальна характеристика засобів комп'ютерних систем та основні визначення

Технічні і програмні засоби комп'ютерних систем (КС) являють собою складні багатофункціональні системи, комплекси технічних і програмних засобів.

Контроль працездатності обладнання здійснюється різними способами. Для цього проводиться періодичний контроль за допомогою спеціальної контрольно-вимірювальної апаратури, вмонтованої апаратури контролю режимів роботи, автоматизованих пристроїв контролю параметрів.

Організація експлуатації КС та мереж (М) вимагає кваліфікованого інженерно-технічного персоналу, добре організованої служби матеріально-технічного постачання, значних експлуатаційних витрат.

Теорія надійності дає змогу вивчати закономірності виникнення пошкоджень і відмов, процесів відновлення працездатності, методів підвищення надійності технічних і програмних засобів.

Теорія експлуатації полягає у вивченні методів забезпечення необхідного рівня надійності й ефективності функціонування засобів у конкретних умовах їх використання.

Експлуатація — це організовані дії технічного персоналу з доведення засобу чи систем до необхідного стану працездатності (чи збереження), підтримання їх у цьому стані і використання з необхідним рівнем ефективності.

Процес експлуатації полягає у виконанні робіт з транспортування, збереження, підготовки до використання, використання, контролю, ремонту та технічного обслуговування (ТО).

Безвідмовність — це властивість обладнання безперервно зберігати працездатність впродовж деякого часу або протягом деякого напрацювання.

Безвідмовність залежить від структури побудови пристрою, конструктивних і технологічних особливостей його виготовлення й умов експлуатації.

Довговічність — це властивість обладнання зберігати працездатність до настання граничного стану з можливими перервами в роботі, пов'язаними з ремонтом і ТО.

Засоби КС та М призначені для довгострокового використання, що забезпечується їх відновлюваністю.

Ремонтопридатність — це властивість засобу, яка характеризується його здатністю до запобігання відмовам чи пошкодженням завдяки регулюванню, виявленню пошкоджень і відновленню працездатності та справності в процесі ремонту чи ТО. Працездатність може відновлюватися в процесі виконання заданих функцій.

Ремонтопридатність закладається в процесі розробки і виготовлення обладнання. Для систем, які довгостроково використовуються, важливою є властивість зберігання їх справного і працездатного стану протягом певного часу використання, зберігання чи транспортування.

Справність — це стан засобу чи системи, за якого він відповідає всім вимогам нормативно-технічної документації (НТД).

Несправність — стан засобу, в якому він не відповідає хоча б одній з вимог НТД.

Працездатність — це стан засобу, в якому він здатний виконувати задані функції, зберігаючи значення основних (не обов'язково усіх) параметрів відповідно до НТД.

Непрацездатність — стан засобу, в якому значення хоча б одного з основних параметрів, які характеризують здатність виконувати задані функції, не відповідає вимогам НТД.

Працездатність і непрацездатність засобу чи системи можуть бути повними чи частковими. Повністю працездатний засіб забезпечує в конкретних умовах максимальну ефективність його застосування. Частково працездатний засіб не забезпечує максимальної ефективності, але її рівень знаходиться в допустимих межах.

Для засобів КС, які можуть функціонувати в деяких можливих станах, характерне поняття ***частково працездатного стану***.

Граничний стан засобу (чи системи) — це стан, в якому його (її) подальше використання за призначенням недопустиме або недоцільне, відновлення його справності чи працездатності неможливе чи недоцільне.

Пошкодження — це подія, що призводить до порушення справності засобу (чи системи) при збереженні ним (нею) працездатності.

Відмова — це подія, котра призводить до порушення працездатності засобу (чи системи).

Відновлювання — процес виявлення й усунення пошкодження чи відмови засобу задля відновлення його працездатності чи справності.

Напрацювання — тривалість чи обсяг роботи засобу.

Технічний ресурс — напрацювання засобу з початку експлуатації до досягнення граничного стану чи капітального ремонту.

Строк служби — календарна тривалість експлуатації засобу від її початку до настання граничного стану незалежно від його напрацювання.

Строк збереженості — календарна тривалість збереження засобу в заданих умовах.

Надійність — властивість засобу зберігати тривалий час в установлених межах значення всіх параметрів, які характеризують здатність виконувати необхідні функції в заданих режимах і умовах використання, ТО, ремонту, зберігання і транспортування. Ця властивість залежно від призначення засобу й умов його використання об'єднує в собі властивості безвідмовності, довгостроковості, ремонтоздатності і збереженості.

Пошкодження і відмови можна класифікувати за рядом ознак. Залежно від характеру виникнення розрізняють пошкодження (і відмови) раптового і поступового характеру.

Відмови раптового характеру характеризуються миттєвою зміною стану одного чи кількох параметрів засобу. Вони проявляються у вигляді поломок, тріщин, обривів, пробоїв конденсаторів, напівпровідникових переходів, транзисторів і інтегральних схем, перегорання резисторів і т.д.

Раптова відмова звичайно виникає внаслідок поступового накопичення несправностей і пошкоджень. Раптовість відмови — явище умовне, тому що вона завжди виникає внаслідок поступового накопичення будь-яких руйнівних дій. Раптовість її виникнення звичайно пов'язана з тим, що параметри даного елемента не контролюються. Якби ми мали досконалі засоби контролю всіх параметрів, усіх елементів системи, то ми спостерігали б тільки відмови поступового характеру. Але оскільки це нереально, то значна частина відмов для нас проявляється раптово.

У комп'ютерних системах важко фіксувати пошкодження поступового характеру, і створюється враження раптовості його виникнення. Чим глибший діагностичний контроль і чим частіше він проводиться, тим більша частка в загальній кількості зареєстрованих пошкоджень апаратури пошкоджень поступового характеру.

Відмови поступового характеру характеризуються зміною одного чи кількох основних параметрів засобу. Поступові пошкодження і відмови, які іноді називають *параметричними*, пов'язані зі зносом деталей, старінням елементів, розрегулюванням пристроїв.

При цьому технічні параметри поступово змінюються, з часом досягають рівня допуску чи переходять через нього, в результаті чого виникає пошкодження чи відмова.

Пошкодження і відмови виникають випадково. Вони можуть бути залежними і незалежними. За взаємозалежністю розрізняють відмови первинні і вторинні. Первинні відмови незалежні, вторинні – залежать від первинних.

Повна відмова призводить до неможливості подальшого використання системи чи засобу за призначенням. У разі виникнення часткової відмови подальше використання системи чи засобу за призначенням можливе, але при цьому значення одного чи кількох основних параметрів знаходяться за межею допусків, і працездатність системи чи засобу знижена. Повні і часткові відмови належать до категорії стійких. На відміну від них перемежаючі відмови виникають багаторазово і самостійно усуваються. В КС та М перемежаючі відмови називають *збоями*.

Залежно від причини виникнення розрізняють конструктивні, виробничі й експлуатаційні відмови (пошкодження). Конструкційна відмова виникає внаслідок недосконалості або порушення правил і норм конструювання засобу. Виробнича відмова — в результаті порушення технології виготовлення або ремонту засобу. Експлуатаційна відмова — в разі порушення правил або умов експлуатації.

2.2.1.2. Експлуатаційно-технічні показники

Показники безвідмовності

До показників безвідмовності належать: середнє напрацювати апаратури на відмову, параметр потоку відмов, інтенсивність відмов і ймовірність безвідмовної роботи.

Середнє напрацювання на відмову — це математичне сподівання випадкового напрацювання апаратури між послідовними відмовами. Зазвичай цей показник використовується в період сталого процесу експлуатації і статистично визначається так:

$$T_{\text{снв}} = \frac{\sum_{i=1}^{n} t_i}{n}, \qquad (2.1)$$

де t_i — відрізок часу безвідмовної роботи між двома послідовними відмовами, $1 < i < n$; n — кількість відмов протягом роботи системи. При цьому

$$t_{\text{р}} = \sum_{i=1}^{n} t_i.$$

де $t_{\text{р}}$ — загальний час роботи системи.

Інтенсивність відмов обчислюється як кількість відмов апаратури за одиницю часу в період її нормальної експлуатації, коли інтенсив-

ність відмов має постійне значення. При цьому статистично вона визначається як $\lambda = \frac{1}{T_{\text{сер}}}$.

Параметр потоку відмов $\omega(t)$ характеризується кількістю відмов за одиницю часу для будь-якого з відрізків часу роботи апаратури, включаючи і періоди нестаціонарної роботи. В стаціонарному режимі $\omega(t) = \lambda = \text{const}$. Для періодів введення системи в нормальну експлуатацію та її фізичного зносу напрацювання між відмовами змінюється за певними законами: для періоду введення в нормальну експлуатацію воно поступово збільшується, а в період фізичного зносу поступово зменшується. Параметр потоку відмов $\omega(t)$, навпаки, спочатку зменшується до значення $\omega(t) = \lambda$, а на заключному періоді експлуатації збільшується.

Імовірність безвідмовної роботи $P(t)$ — це ймовірність того, що протягом заданого напрацювання відмова засобу не відбудеться за умови його працездатності в початковий момент. Математично вона найчастіше оцінюється так:

$$P(t) = \exp\left(-\int_0^t \omega(u)\,du\right).$$

При $\omega(t) = \lambda$ отримуємо $P(t) = \exp(-\lambda t)$.

Якщо $t = 0$, то $P(t) = 1$; при $t = T_0$ одержуємо $P(t) = 0{,}37$; при $t = \infty$ — $P(t) = 0$.

Аналогічно можна визначити показники роботи без виникнення пошкоджень: напрацювання апаратури на пошкодження, інтенсивність пошкоджень, параметр потоку пошкоджень, імовірність роботи апаратури без пошкоджень.

Технічні засоби КС та М мають відносно високу надійність. Для багатьох з них повна відмова може статися з досить малою ймовірністю і виникає дуже рідко, хоча пошкодження з'являються значно частіше. В одних випадках пошкодження не впливає на ефективність їх роботи, якщо використовується резервне обладнання або в момент пошкодження немає нагальної потреби в даній системі. В інших випадках пошкодження може негативно вплинути на ефективність роботи системи, зменшивши її частково, або зірвавши виконання призначених функцій.

Показники відновлюваності

Показниками відновлюваності обладнання КС та М є середній час, необхідний для відновлення працездатності $i_{\text{в}}$; інтенсивність

відновлення μ; параметр закону розподілу ймовірності відновлення засобу $\nu(t)$; ймовірність відновлення працездатності $V(t)$.

Середній час відновлення працездатності апаратури статистично визначається так:

$$t_{\text{в}} = \frac{\sum_{i=1}^{n} t_{\text{в}_i}}{n},$$

де n — кількість відмов чи пошкоджень апаратури на відрізку роботи; $t_{\text{в}}$ — час відновлення працездатності апаратури після виникнення i-ї відмови чи пошкодження.

Інтенсивність відновлення характеризується кількістю відновлень працездатності апаратури за одиницю часу в період нормальної експлуатації системи і дорівнює $\mu = \frac{1}{t_{\text{в}}}$. Параметр закону розподілу часу відновлення апаратури $\nu(t)$ визначається кількістю відновлень апаратури за одиницю часу в будь-який період роботи системи. В період нормальної експлуатації $\nu(t) = \mu = \text{const}$.

Імовірність того, що за певний час t буде відновлена працездатність апаратури, дорівнює

$$V(t) = 1 - \exp\left(-\int_0^t \nu(u)\,du\right);$$

$V(t) = e^{-\mu t}$ при $V(t) = \mu$;

$V(t) = 0$ при $t = 0$;

$V(t) = 0{,}63$ при $t = t_{\text{в}}$;

$V(t) = 1$ при $t = \infty$.

Показники довговічності і збереженості

До показників довговічності належать:

- гамма-процентний ресурс — напрацювання засобу чи системи, протягом якого вони не досягають граничного стану із заданою ймовірністю $1 - \gamma$;
- призначений ресурс — сумарне напрацювання засобу чи системи, після якого експлуатація засобу чи системи має бути припинена незалежно від його стану;
- середній ресурс до списання — середній ресурс від початку експлуатації до списання;
- середній ресурс до капітального ремонту — середній ресурс від початку експлуатації до першого капітального ремонту; гамма-

процентний строк служби — строк служби, протягом якого засіб чи система не досягнуть граничного стану з імовірністю $P = 1 - \gamma$;

- середній міжремонтний строк служби — середній строк служби між суміжними капітальними ремонтами засобу чи системи; середній строк служби до капітального ремонту — середній строк служби від початку експлуатації до першого капітального ремонту;
- середній строк служби до списання – середній строк служби від початку експлуатації засобу чи системи до списання.

Збереженість засобу чи системи характеризується:

- гамма-процентним строком збереженості – тривалістю збереження, протягом якого у засобу чи в системі зберігаються встановлені показники з заданою ймовірністю $1 - \gamma$;
- середнім строком збереженості — математичним сподіванням строку збереженості.

Довговічність може обмежуватися технічним або моральним ресурсом. Технічний ресурс визначається фізичним станом засобу чи системи під впливом робочого навантаження, метеорологічних умов, а іноді і недостатньо кваліфікованої експлуатації. Моральний ресурс залежить від якості розробки і виготовлення засобу чи системи, а також від прогресу розвитку науки і техніки, зміни вимог до засобів і систем з часом, відповідності засобів і систем цим вимогам.

Комплексні показники

Комплексні показники називають експлуатаційними коефіцієнтами. Вони враховують поняття безвідмовності і відновлюваності засобів і систем, особливості їх використання і технічної експлуатації.

До таких показників належать:

1) коефіцієнт готовності $K_г$, який характеризує час, протягом якого будь-якої миті можна вважати засіб чи систему готовими до використання за призначенням:

$$K_г = \frac{T_р}{T_р + t_{вс}} \le 1,$$

де $T_р$ — тривалість часу роботи засобу чи системи в справному стані; $t_{вс}$ — сумарні витрати часу на відновлення працездатності засобу чи системи за період експлуатації $t = T_р + t_{вс}$;

2) коефіцієнт простою $K_п$. який характеризує час, протягом якого будь-якої миті можна вважати засіб чи систему неготовими до використання:

$$K_п = 1 - K_г = \frac{t_{вс}}{T_р + t_{вс}} \le 1;$$

3) функція готовності, яка характеризує залежність коефіцієнта готовності від тривалості експлуатації:

$$K_г(t) = l^{-F(t)}\left(\eta + \int_0^t \nu(u) e^{F(u)} du\right), \quad F(t) = \int \left(\omega(u) + \nu(u)\right) du,$$

де η — імовірність того, що в момент початку використання засобу чи системи при $t = 0$ вони будуть в справному стані.

При експоненціальних законах розподілу часу безвідмовної роботи і часу відновлення

$$\omega(t) = \lambda, \ \nu(t) = \mu;$$

$$K_г(t) = \frac{\mu}{\lambda + \mu} + \left(\eta - \frac{\mu}{\lambda + \mu}\right) e^{-(\lambda + \mu)t}.$$

Якщо при $t = 0$ $\eta = 1$, то

$$K_г(t) = \frac{\mu}{\lambda + \mu} + \frac{\mu}{\lambda + \mu} e^{-(\lambda + \mu)t}.$$

У випадку, коли при $t = 0$ $\eta = 0$, одержуємо

$$K_г(t) = \frac{\mu}{\lambda + \mu}\left(1 - e^{-(\lambda + \mu)t}\right).$$

Функція простою при цьому має такий вигляд:

$$K_г(t) = \frac{\lambda}{\lambda + \mu} + \left(\eta - \frac{\mu}{\lambda + \mu}\right) e^{-(\lambda + \mu)t};$$

4) коефіцієнт оперативної готовності

$$K_{ог}(t) = K_г P(t_{вз}),$$

де $P(t_{вз})$ — імовірність виконання завдання засобом чи системою за встановлений для цього час $t_{вз}$;

5) функція оперативної готовності

$$K_{ог}(t) = K_г(t) P(t_{вз});$$

6) коефіцієнт технічного використання

$$K_{тв} = \frac{T_р}{T_р + t_{вс} + t_{обсл.с}} \le K_г,$$

де $t_{обсл.с}$ — сумарний час технічного обслуговування (ТО) засобу чи системи за час експлуатації $t = T_р + t_{вс} + t_{обсл.с}$. Він характеризує час, протягом якого з урахуванням простою на ремонті і ТО засіб чи система готові до використання;

7) коефіцієнт збереження ефективності $K_{зб.е}$, який характеризується відношенням ефективності системи з урахуванням реального стану її надійності до ефективності абсолютно надійної системи згідно з вибраними критеріями оцінки ефективності. На практиці $K_{зб.е} < 1$ і завдання служби експлуатації полягає у його максимальному збільшенні;

8) коефіцієнт використання засобу чи системи $K_в$, який характеризує час, протягом якого засіб чи система використовується за призначенням.

Питання для самоперевірки

1. Що являє собою технічна експлуатація засобів КС та М?
2. Які основні види робіт виконуються в процесі технічної експлуатації КС та М?
3. Що таке надійність роботи?
4. Чим відрізняються поняття справності і працездатності засобів КС та М?
5. Який зміст мають поняття пошкодження і відмови?
6. Як класифікуються відмови і пошкодження КС та М?
7. Перелічіть основні експлуатаційно-технічні показники технічних і програмних засобів?
8. Назвіть показники безвідмовності, відновлюваності та довговічності?
9. Перелічіть комплексні експлуатаційні показники.
10. Який вираз мають основні експлуатаційні показники?

2.2.2. Організація технічного обслуговування КС та М

2.2.2.1. Загальні положення

У комплекси технічних засобів КС та М входить різноманітне обладнання: електронне, радіоелектронне, електротехнічне, електромеханічне, механічне тощо. Радіоелектронне обладнання побудоване в основному на базі цифрових і аналогових інтегральних схем

другого й третього ступенів інтеграції. Широко використовується поелементне, апаратурне, інформаційне, часове і функціональне резервування.

Контроль працездатності обладнання здійснюється різними способами:

— за наявністю необхідної інформації;

— за результатами періодичного контролю з використанням спеціальної контрольно-вимірювальної апаратури;

— за даними вбудованої апаратури контролю режимів роботи;

— за показниками автоматизованої системи контролю параметрів.

В КС та М використовується, крім того, тестовий контроль працездатності ЕОМ у ряду цифрових пристроїв. Значну частину вузлів і агрегатів КС та М неможливо систематично контролювати.

Застосування в КС та М цифрових і інтегральних схем з високим ступенем інтеграції елементів приводить до того, що діагностичний контроль обладнання КС та М втрачає інформативність, однак відкриваються широкі можливості для тестового контролю його працездатності. У такому випадку важко фіксувати відмови поступового характеру і створюється враження раптовості виникнення відмов. Чим з меншою кількістю елементів або технічних параметрів роблять діагностичний контроль і чим рідше він проводиться, тим менш виявляється частка відмов поступового характеру в загальній кількості зареєстрованих відмов апаратури.

Практична відсутність контролю механічного, електромеханічних і ряду радіотехнічних вузлів і агрегатів обладнання КС та М також не дає змоги діагностувати їх функціонування. Виникаючі в них відмови при цьому також реєструються як раптові, хоча процеси зносу і протікають протягом досить тривалого часу.

Велика вартість обладнання КС та М, висока продуктивність і висока відповідальність розв'язуваних ними завдань зумовлюють значні соціальні й економічні втрати при їхньому простої унаслідок виникаючих відмов або погано організованої експлуатації. Тому технічному обслуговуванню (ТО) обладнання КС та М надається велике значення.

Технічне обслуговування являє собою операцію або комплекс операцій з підтримки працездатності або справності виробу в процесі використання його за призначенням, чекання використання, збереження і транспортування.

Найбільш тривалим є етап використання виробу (системи, об'єкта) за призначенням. Тому проведенню ТО на цьому етапі надається велике значення. Обладнання КС та М, не використовуване в даний час, що перебуває в резерві або в режимі чекання використання, та-

кож має потребу в постійному ТО, оскільки воно при цьому може зазнати впливу навколишнього середовища. Ще більшою мірою середовище впливає на обладнання під час транспортування і зберігання, тому що тут не виключені ще й механічні впливи. Тож і на цих етапах має проводитися контроль стану обладнання і необхідне ТО.

Розрізняють такі стратегії ТО:

— регламентоване;

— за станом;

— періодичне;

— комбіноване — з використанням кількох стратегій його проведення.

Технічне обслуговування обладнання проводиться з визначеною періодичністю. Під періодичністю ТО розуміють інтервал часу або наробітку між даним видом ТО і наступним таким самим або більш складним видом ТО. Звичайно ТО складних систем проводиться з поділом робіт (обладнання) за ступенями складності. Наприклад, при календарному обслуговуванні обладнання один раз на рік забезпечується повне обслуговування всіх агрегатів, блоків, плат і вузлів. Деякі вузли й агрегати не потребують більш частого обслуговування. Однак іншим необхідне частіше ТО. Такі вузли й агрегати поділяються на кілька груп і на підставі досвіду експлуатації, що накопичується для даної системи протягом одного—двох років або досвіду експлуатації подібних систем (комплексів, агрегатів), установлюють для кожної групи максимальну періодичність ТО. Далі такі групи (комплекси) агрегатів поділяють на підгрупи (блоки, плати), що вимагають ще більш частого ТО.

Зазвичай проводиться оперативний контроль стану обладнання системи і за його результатами — оперативне ТО. Оперативний контроль працездатності обладнання передбачає швидке виконання операцій перевірки працездатності об'єкта в процесі його функціонування.

Технічне обслуговування проводиться у визначеному обсязі. Обсяг ТО передбачає визначену сукупність операцій ТО і трудомісткість їх виконання.

У процесі експлуатації обладнання КС та М може бути в різному технічному стані, під яким розуміють сукупність змін властивостей обладнання в процесі експлуатації, що характеризується в даний момент ознаками, встановленими нормативно-технічною документацією. Для кожного об'єкта нормативно-технічною документацією визначається перелік таких станів (справність, працездатність, ушкодження тощо).

Організація і проведення ТО — це система, що являє собою сукупність таких взаємозалежних складових, як технічний персонал,

засоби у документації, необхідні для підтримки й відновлення працездатності об'єктів у КС та М.

Система ТО характеризується визначеними показниками ефективності, що відбивають її здатність виконувати функції з підтримки справності або працездатності обладнання при визначених витрат часу, трудових ресурсів і матеріальних засобів. Удосконалювання системи ТО сприяє не тільки підвищенню надійності, а й довговічності обладнання.

2.2.2.2. Стратегії і форми технічного обслуговування

Чим складніша технічна система, тим більше значення надається її ТО. Процес ТО має забезпечувати не тільки високу готовність і безвідмовність обладнання, а й його довговічність. У процесі експлуатації КС та М інженер повинен постійно вишукувати можливості продовження технічного ресурсу обладнання. Особливу увагу слід приділяти нормуванню витрат ресурсу обладнання і трудових ресурсів, електроенергії, пально-мастильних матеріалів і елементної бази. Експлуатацію КС, М й іншого обладнання необхідно організувати так, щоб вона була гранично ощадливою.

Досягненню цих цілей у процесі експлуатації сприяють ТО, ремонт техніки, її доробка, оптимізація режимів роботи, впровадження апаратури автоматизованого тестового та діагностичного контролю, підвищення кваліфікації технічного персоналу, забезпечення своєчасного поповнення запасу елементів та інші заходи.

Організація ТО припускає визначення стратегії, методу, видів, періодичності, обсягу й технології його проведення, потреби в обслуговуючому персоналі.

У процесі експлуатації КС та М використовуються такі стратегії ТО:

1. Регламентоване ТО, передбачене нормативно-технічною документацією і виконуване з періодичністю та в обсязі, встановленими даною документацією, незалежно від технічного стану обладнання в момент початку обслуговування. Цій стратегії відповідають два види ТО — періодичне і регламентоване.

За періодичного ТО регламентується тільки періодичність його проведення, а обсяг робіт визначається станом обладнання до моменту початку обслуговування.

Регламентоване ТО передбачає планово-запобіжну заміну елементів (вузлів, блоків, агрегатів) після вироблення призначеного їм ресурсу або календарного терміну служби, незалежно від наробітку апаратури.

Регламентоване ТО передбачає проведення робіт профілактичного характеру з періодичністю, зумовленою наробітком й ресурсом обслуговування. Періодичність ТО може встановлюватися за принципами календарної та сезонної регламентації. Обсяг ТО передбачається нормативно-технічною документацією.

Календарні й сезонні регламентовані ТО передбачають такі види обслуговування, періодичність яких визначається за календарним принципом. Профілактична заміна елементів здійснюється з огляду на термін роботи, незалежно від того, були вони в робочому режимі або в режимі чекання використання. Ця стратегія ТО застосовується при експлуатації систем або агрегатів, які зазнали метеорологічних впливів, а також систем на збереженні чи з тривалими перервами у роботі.

Регламентоване за наробітком ТО пропонує призначення термінів і обсягів проведення обслуговування відповідно до ресурсу використовуваних в апаратурі елементів. Така стратегія зазвичай застосовується для ТО апаратури або її агрегатів, не охоплених системою контролю і тих, що перебувають в режимі безперервної роботи.

Обидва ці різновиди стратегії ТО вимагають значних витрат часу на обслуговування, вимушені простої апаратури, великих витрат елементної бази, заміну якої необхідно робити відповідно до виробленого ресурсу або терміну незалежно від її технічного стану.

Отже, ця стратегія не забезпечує максимального рівня надійності обладнання і недостатньо економна.

2. Технічне обслуговування за станом — безперервний або періодичний контроль технічного стану, працездатності або рівня надійності обладнання. Цій стратегії відповідають три види ТО: безперервний контроль параметрів; періодичний контроль параметрів; контроль рівня надійності.

В устаткуванні визначається основна діагностична ознака або ряд основних параметрів, контроль яких практично цілком характеризує його працездатність або технічний стан. Рішення про проведення ТО приймається за результатами контролю параметрів. Однак, якщо при контролі ці параметри відповідають установленим нормам, жодні роботи з ТО не проводяться. При цьому допускається виникнення ушкодження обладнання за наявності структурного резерву.

Обслуговування за рівнем надійності передбачає експлуатацію обладнання до відмови без проведення ТО. За рівнем параметра потоку відмов або іншої характеристики надійності приймається рішення про проведення тих чи інших робіт з підвищення надійності обладнання, якщо в цьому є необхідність.

У таких складних системах як КС та М або на устаткуванні, що входить до складу їхніх комплексів технічних засобів, зазвичай проводиться комбіноване ТО, що включає елементи як регламентованого обслуговування, так і ТО за станом для різних систем або агрегатів обладнання. У такий спосіб слід обслуговувати найбільш відповідальні частини складних систем.

Від застосовуваних стратегій і видів ТО залежать багато характеристик системи, тому їх вибір повинен бути добре обґрунтованим.

3. Періодичне ТО здійснюється в обсязі, зумовленому технічним станом апаратури на момент початку обслуговування, періодичність якого регламентується. Широко використовується поетапний метод обслуговування з розподіленою трудомісткістю, за якого час проведення чергового циклу ТО узгоджується з можливістю вимикання апаратури внаслідок її незайнятості при виконанні поставлених завдань. Наявність систем автоматизованого контролю технічного стану і працездатності обладнання дає змогу використовувати більш економну стратегію ТО за станом. Її перевага полягає в можливості забезпечення більш високого рівня надійності апаратури і вищій економності. За цієї стратегії ТО обслуговуючий персонал зайнятий тільки найнеобхіднішими операціями і значно скорочуються витрати запасних елементів.

Проведення ТО дає змогу зменшувати кількість відмов поступового характеру і запобігати певній частині відмов. ТО доцільно проводити на устаткуванні, фізично зношеному або старому.

Обслуговування за рівнем надійності доцільно для систем, яким властиві відмови раптового характеру. Переведення сучасної апаратури на цифрову елементну базу з високим ступенем інтеграції елементів привів, з одного боку, до значного підвищення її надійності, а з іншого — до значного зменшення частки відмов поступового характеру в загальному потоці відмов, що значно зменшує доцільність проведення ТО.

Широкому впровадженню обслуговування за рівнем надійності сприяє побудова високонадійних систем при використанні структурного резервування на базі систем автоматичного керування конфігурацією системи. Цей вид ТО найбільш економний, оскільки за його застосування витрачається мінімум запасних елементів.

Унаслідок того, що складні системи, до яких належить й обладнання КС та М, містять дуже велику кількість різнотипних цифрових і аналогових елементів, що часто сполучаються з механічними вузлами, а в процесі експлуатації вони нерідко зазнають метеорологічних впливів, впливів нестабільності напруги електроживлення, перехідних процесів при вмиканні і вимиканні, змін навантаження в

роботі, сумарний потік відмов має складові як поступових, так і раптових відмов. Значна частина обладнання в таких системах виявляється неконтрольованою. Тому для цих систем знаходять широке застосування періодичні ТО.

У процесі обґрунтування періодичності ТО може бути кілька підходів залежно від обраного визначального параметра. Як такі параметри можуть бути забезпечення не нижче мінімально припустимого рівня ймовірності безвідмовної роботи обладнання, максимального коефіцієнта готовності або коефіцієнта технічного використання, максимуму економічних показників використання обладнання тощо.

Залежно від зайнятості технічного персоналу розрізняють потоковий, централізований і децентралізований методи ТО.

2.2.2.3. Періодичність та обсяг технічного обслуговування

Для організації ТО дуже важливе визначення термінів і обсягів його проведення. Періодичність і глибина ТО — поняття тісно зв'язані. Кількість відмов у системі за час її експлуатації залежить від кількості її елементів, надійності цих елементів, умов експлуатації, якості ТО і ремонту. Якість ТО визначається конструктивними особливостями апаратури і рівнем кваліфікації обслуговуючого технічного персоналу. Воно залежить також від періодичності і глибини ТО. Під глибиною ТО розуміється частка елементів або вузлів апаратури, що контролюються та обслуговуються в межах даного циклу ТО. Для більш простих систем глибина ТО вище, ніж для обладнання КС та М, глибина ТО якого зазвичай становить менше 100 %.

Чим більшу кількість елементів системи буде перевірено в процесі ТО, тим рідше можна проводити цикли ТО, грунтуючись на ресурсі використовуваних елементів і вузлів, ступені їхнього фізичного зносу в процесі експлуатації системи в конкретних умовах.

При обґрунтуванні періодичності ТО визначальний параметр вибирається відповідно до призначення системи і вимог до неї, особливостей конструкції, рівня надійності й умов експлуатації.

Для оцінки надійності серед експлуатаційних показників технічних систем використовуються методи аналітико-статистичного та статистичного моделювання. Вони не виключають, а доповнюють один одного завдяки тому, що недоліки одного компенсуються перевагами іншого.

При аналітико-статистичному методі найчастіше припускають, що потік відмов або ушкоджень апаратури — найпростіший, а час її відновлення підлягає експонентному законові розподілу. Практика

експлуатації підтверджує справедливість таких припущень, однак не завжди. Якщо припустити найпростішим потік випадкових подій, можна одержати аналітичне рішення задачі. Більш того, часто, замінивши потік будь-якої структури найпростішим з тією самою щільністю, можна отримати задовільні щодо точності результати.

При розгляді всього періоду експлуатації обладнання можна зазначити три характерні етапи. Перший етап — припрацьовування апаратури, етап уведення її у роботу. Він характеризується сильно змінюваним у часі параметром потоку відмов або ушкоджень і залежністю від часу параметра закону розподілу часу відновлення апаратури $\nu(t)$. Це не дає змоги на даному етапі використовувати чисто марківську модель функціонування системи. Другий етап — нормальна експлуатація, протягом якої параметр потоку відмов $\omega(t) = \lambda$ і параметр закону розподілу часу ремонту мало міняються в часі. Тут марківські моделі досить добре описують процес функціонування обладнання КС та М. Далі настає третій етап — фізичний знос апаратури, де параметр потоку відмов починає збільшуватися в часі. У загальному потоці відмов переважають відмови поступового характеру і припущення, справедливі для марківської моделі, не виконуються.

Дослідження показують, що для складних систем із великою кількістю різноманітних елементів (вузлів, агрегатів), у період нормальної експлуатації практичну незалежність параметра потоку відмов від часу можна забезпечити тільки періодичним проведенням ТО. У результаті ТО вдається значно знизити відсоток відмов поступового характеру, і тоді відмови носять в основному раптовий характер й мало залежать від часу експлуатації. Цього можна домогтися для багатьох систем тільки завдяки правильно організованому ТО.

Технічне обслуговування обладнання проводиться відповідно до встановлених форм.

Форма ТО-1 розглядається як оперативне ТО і зводиться до контролю основних параметрів і засобів КС і М та проведення робіт зі своєчасної підготовки засобів до забезпечення польотів. ТО-1 полягає в перевірці загальної працездатності обладнання.

Звичайно оперативне ТО проводиться щодня. Але оперативний контроль за необхідності може здійснюватись і частіше, наприклад, під час приймання-передавання зміни чергування на об'єкті. Результати оперативного контролю можуть вимагати проведення визначеного оперативного ТО за формою ТО-1.

Форма ТО-2 передбачає щотижневе ТО або ТО після наробітку 170 год. ТО-2 проводиться оперативними бригадами ремонтно-експлуатаційних майстерень або черговим персоналом об'єкта.

Трудомісткі форми ТО виконуються з визначеною періодичністю. Ці форми ТО характеризуються досить великим обсягом робіт порівняно з оперативним ТО. В основу трудомістких видів ТО покладена регламентація за наробітком або за календарним принципом.

Форма ТО-3 відповідає проведенню щомісячних робіт або робіт через 750 год наробітку.

Форма ТО-4 передбачає визначений вид робіт, які проводяться щокварталу або через 2250 год наробітку.

Відповідно до форми ТО-5 проводиться щопівріччне ТО або після наробітку 4500 год.

Визначений вид робіт на складних системах проводиться один раз на рік після закінчення 8800 год наробітку (форма ТО-6).

Допускається відхилення термінів початку робіт з ТО на $\pm 15\,\%$ від установленої періодичності. Тривалість ТО залежить від організації робіт, їх технології, наявності спеціальних засобів для проведення, необхідного технічного персоналу та ряду інших чинників.

Відомі декілька підходів до визначення періодичності ТО у разі використання стратегії регламентованого ТО напрацюванням засобу обслуговування, в яких обумовлюється, що потоки відмов і відновлювання працездатності обладнання являють собою нестаціонарні пуассонівські потоки з параметрами $\omega(t)$ і $\nu(t)$ відповідно.

Нестаціонарний пуассонівський потік — це ординарний потік випадкових подій без післядії, для якого в будь-який момент часу t існує кінцевий параметр $\omega(t)$. Розрізняють потоки за змінними детермінованими і випадковими параметрами. У першому випадку функції $\omega(t)$ і $\nu(t)$ заздалегідь відомі і їх значення залежать тільки від моменту часу t, у другому випадку $\omega(t)$ і $\nu(t)$ — випадкові функції.

Модель нестаціонарного пуассонівського потоку за перемінними параметрами дає змогу при відповідному виборі залежності досить добре описати реальний нестаціонарний потік відмов або ушкоджень апаратури за обліку старіння або внесення її елементів.

Розглянемо кілька підходів до визначення періодичності ТО. Нехай вимоги до надійності системи задані у вигляді мінімально припустимого значення ймовірності $P_{зп}$ перебування її в справному стані до початку чергового циклу планового ТО. Припустимо далі, що після виконання чергового ТО система повертається у вихідний стан, час відновлення підлягає експоненціальному законові розподілу з параметром $\nu(t) = \mu_{const}$, а тривалість ТО не враховується. Для різних систем характер зміни функції $\omega(t)$ різний. Часто можна прийняти, що

$$\omega(t) = a + bt,$$

де a — параметр потоку раптових відмов; b — швидкість зміни параметра поступових відмов.

Імовірність перебування системи в справному стані в момент часу $T_п$ початку наступного циклу ТО

$$P(T_п) = e^{-F(T_п)}\left(1 + \int_0^{T_п} \mu(u) e^{F(u)} du\right),$$

де

$$F(T_п) = \int_0^{T_п} (\omega(u) + \nu(u))\, du.$$

Якщо параметр потоку відмов не залежить від часу $\omega(t) = \lambda$, то необхідна періодичність ТО визначається з нерівності:

$$P(T_п) = e^{-(\lambda+\mu)T_п}\left(1 + \frac{\mu}{\lambda+\mu}\left(e^{(\lambda+\mu)T_п} - 1\right)\right) \geq P_{зп};$$

$$F(T_п) = (\lambda + \mu)T_п.$$

Звідси

$$T_п \leq -\frac{1}{\lambda+\mu} \ln \frac{P_{зп}(\lambda+\mu) - \mu}{\lambda}. \qquad (2.2)$$

У виразі (2.2) імовірність перебування системи в справному стані

$$P_{зп} \geq \frac{\mu}{\lambda+\mu}.$$

Для відновлюваних у проміжках між циклами ТО систем (при $\mu = 0$)

$$T_п \leq -\frac{1}{\lambda} \ln P_{зп}.$$

Якщо параметр потоку відмов лінійно залежить від часу $\omega(t) = a + bt$, тоді ймовірність:

$$P(T_п) = e^{-F(T_п)}\left(1 + \int_0^{T_п} \mu\, e^{F(u)} du\right) \geq P_{зп}; \qquad (2.3)$$

де

$$F(T_п) = (a+\mu)T_п + \frac{1}{2} b T_п^2.$$

Розв'язавши нерівність (2.3) відносно $T_{\text{п}}$, дістанемо необхідну періодичність планового ТО $T_{\text{об}}$.

Якщо в проміжках між циклами ТО обладнання не відновлюється ($\mu = 0$), то періодичність циклів технічного обслуговування $T_{\text{об}}$ буде задовольняти:

$$T_{\text{об}} \le -\frac{a}{b} + \sqrt{\frac{a^2}{b^2} - \frac{2}{b}\ln P_{\text{зп}}}.$$

У процесі експлуатації необхідно забезпечити максимальний коефіцієнт технічного використання обладнання за призначенням. При цьому слід враховувати його простої як на відновлення працездатності, так і при проведенні ТО.

Коефіцієнт технічного використання обладнання виражається так:

$$K_{\text{тв}} = \frac{lT_{\text{сер}}}{l(T_{\text{сер}} + t_{\text{в}}) + t_{\text{о}}} = \frac{lT_{\text{сер}}}{T_{\text{о}}(1 + \omega_{\text{сер}} t_{\text{в}})}, \qquad (2.4)$$

де l — середня кількість відмов апаратури за час $T_{\text{п}}$ між двома послідовно проведеними циклами ТО, $l = \omega_{\text{сер}} K_{\text{тв}} = T_{\text{об}}$; $T_{\text{сер}} = \dfrac{1}{\omega_{\text{сер}}}$ — середній наробіток апаратури на відмову за розглянутий період експлуатації $t_{\text{р}} = nT_{\text{об}}$; $t_{\text{в}}$ — середній час відновлення; $t_{\text{о}}$ — тривалість проведення одного циклу ТО; $T_{\text{об}}$ — періодичність проведення ТО за ресурсом; n — кількість циклів ТО, що проводяться.

При $t_{\text{р}} = nT_{\text{об}}$ і $\omega(t) = a + (i-1)(i-k)bT_{\text{об}} + bt$ (рис. 2.3, *а*)

$$\omega_{\text{сер}} = a + \frac{1}{2}kbT_{\text{об}} + \frac{1}{2}(1-k)bnT_{\text{об}}, \qquad (2.5)$$

де k — глибина проведеного ТО ($0 \le k \le 1$); $0 \le t \le T_{\text{п}}$.

Підставивши отримане відповідно до формули (2.5) значення $\omega_{\text{сер}}$ у вираз (2.4) і дослідивши отриману функцію на екстремум відносно $T_{\text{об}}$, одержимо формулу для визначення оптимальної періодичності ТО:

$$T_{\text{об.opt}} = \sqrt{\frac{2t_{\text{о}}}{bt_{\text{в}}\left(k + n(1-k)\right)}}. \qquad (2.6)$$

Звідси при $k = 1$ (рис. 2.3, *б*) можна дістати вираз:

$$T_{\text{об.opt}} = \sqrt{\frac{2t_{\text{о}}}{bt_{\text{в}}}}. \qquad (2.7)$$

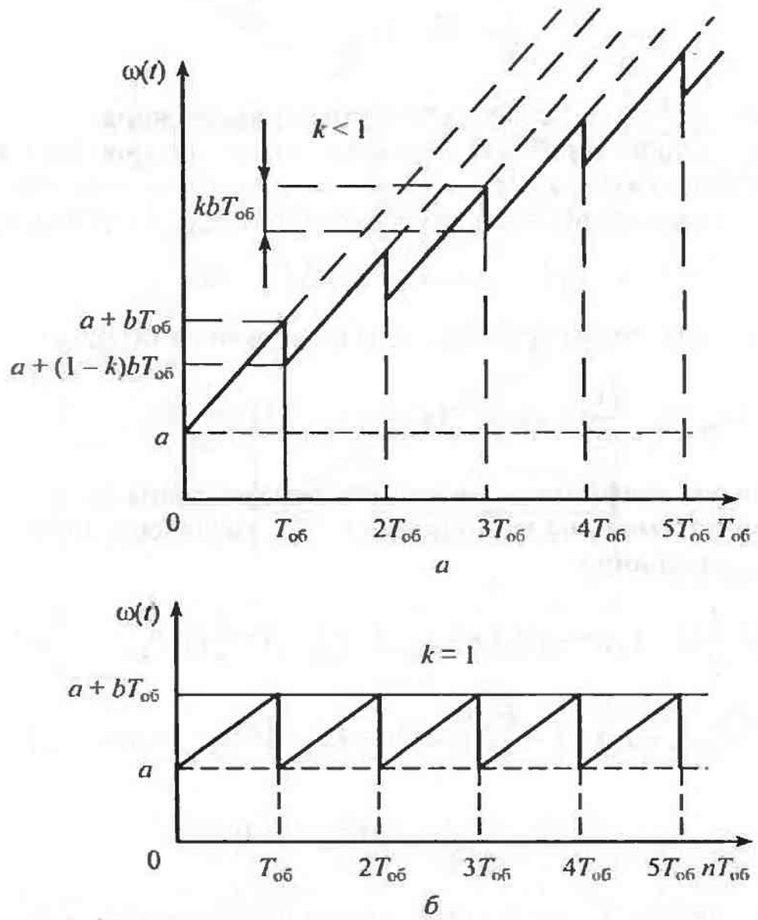


Рис. 2.3. Залежності параметра потоку відмов апаратури, що обслуговується, за різних глибин ТО:
а — при глибині $k < 1$; *б* — при глибині $k = 1$

Якщо швидкість зміни параметра поступових відмов між послідовно проведеними циклами ТО різна і дорівнює b_i ($0 \le i \le n$), то за постійної глибини ТО k середнє значення параметра потоку відмов за час $nT_{\text{об}}$ дорівнює:

$$\omega_{\text{сер}} = a + \frac{1}{n}(1-k)T_{\text{об}} \sum_{m=1}^{n} \sum_{j=1}^{m=1} b_j + \frac{1}{2n} T_{\text{об}} \sum_{m=1}^{n} b_m.$$

При цьому максимальне значення коефіцієнта технічного використання може бути отримане при

$$T_{\text{об.}opt} = \sqrt{\frac{2nt_{\text{о}}}{(1-k)t_{\text{в}}(2\sum_{m=1}^{n}\sum_{j=1}^{m=1} b_j + \sum_{m=1}^{n} b_m)}}.$$

Звідси при $b_i = b = \text{const}$ може бути отримано вираз (2.6).

Якщо до проведення ТО параметр потоку відмов $\omega(t)$ підкоряється законові $\omega(t) = a + bt + ct^2$, де a — параметр раптових відмов, $(bt + ct^2)$ — параметр відмов поступового характеру, то після n циклів ТО

$$\omega_n(nT_{\text{об}}) = a + (1-k)n(bT_{\text{об}} + cT_{\text{об}}{}^2).$$

Середнє значення параметра потоку відмов на інтервалі $(0, T_{\text{п}})$

$$\omega_{\text{сер}} = a + \frac{1}{2}(1-k)(n-1)(bT_{\text{об}} + cT_{\text{об}}{}^2) + \frac{1}{2}bT_{\text{об}} + \frac{1}{3}cT_{\text{об}}^2.$$

Максимум коефіцієнта технічного використання може бути здобутий при оптимальній періодичності ТО, визначеній рішенням відносно $T_{\text{об.}opt}$ рівняння:

$$T_{\text{об.}opt}\left(1 + \frac{1}{2}(1-k)(n-1)t_{\text{в}}\left(bT_{\text{об.}opt} + cT_{\text{об.}opt}^2\right) + \frac{1}{2}bt_{\text{в}}T_{\text{об.}opt} + \frac{1}{3}cT_{\text{об.}opt}^2\right) -$$

$$-\left(T_{\text{об.}opt} - t_{\text{о}}\right)\left(1 + \frac{1}{2}(1-k)(n-1)t_{\text{в}}\left(2bT_{\text{об.}opt} + 3cT_{\text{об.}opt}^2\right) + \right.$$

$$\left. + bT_{\text{об.}opt} + cT_{\text{об.}opt}^2\right) = 0.$$

Якщо глибина ТО на всьому інтервалі експлуатації $k = 1$, то для визначення $T_{\text{об.}opt}$ необхідно розв'язати рівняння:

$$\frac{2}{3}ct_{\text{в}}T_{\text{об.}opt}^3 + \left(\frac{1}{2}bt_{\text{в}} - ct_{\text{о}}t_{\text{в}}\right)T_{\text{об.}opt}^2 - bt_{\text{о}}t_{\text{в}}T_{\text{об.}opt} - t_{\text{о}} = 0.$$

Залежності оптимальної періодичності ТО від середнього часу відновлення $t_{\text{в}}$ і середньої тривалості циклу ТО $t_{\text{о}}$ наведено на рис. 2.4.

Залежності оптимальної періодичності ТО від тривалості експлуатації (кількості циклів ТО n) при квадратичній залежності $\omega(t)$ наведено на рис. 2.5, *а*. Для забезпечення сталості середнього значення наробітку на відмову протягом усього періоду експлуатації необхідно скорочувати інтервали між сусідніми циклами ТО при збільшенні тривалості експлуатації.

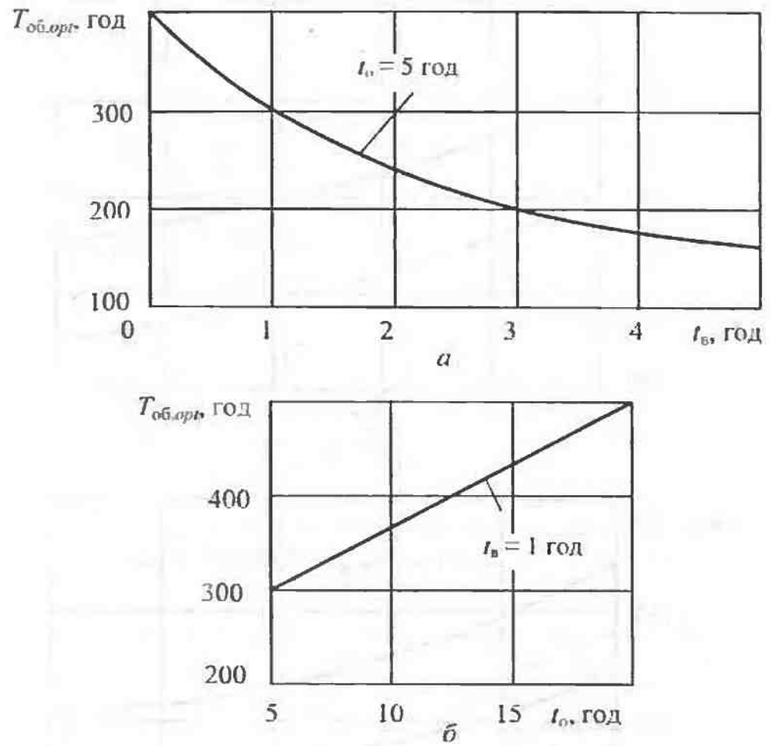


Рис. 2.4. Залежності оптимальної періодичності ТО від середнього часу відновлення апаратури (*а*) і середньої тривалості циклу ТО (*б*) при $a = 10^{-3}$ год$^{-1}$, $b = 10^{-5}$ год$^{-2}$, $c = 10^{-7}$ год$^{-3}$

Слід звернути увагу, що зі збільшенням тривалості експлуатації (кількості циклів ТО n) при постійній періодичності ТО середнє значення наробітку на відмову між сусідніми циклами ТО зменшується (рис. 2.5, *б*).

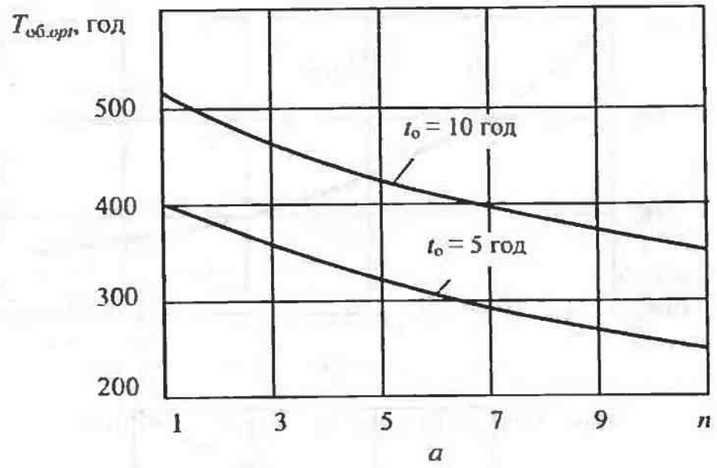
При лінійному законі зміни в часі параметра потоку відмов $\omega(t)$ і $b_i = b$ проведення n циклів ТО за наробітком дає змогу продовжити ресурс системи в Δ раз:

$$\Delta = \sqrt{\frac{n}{k + (1-k)n}},$$

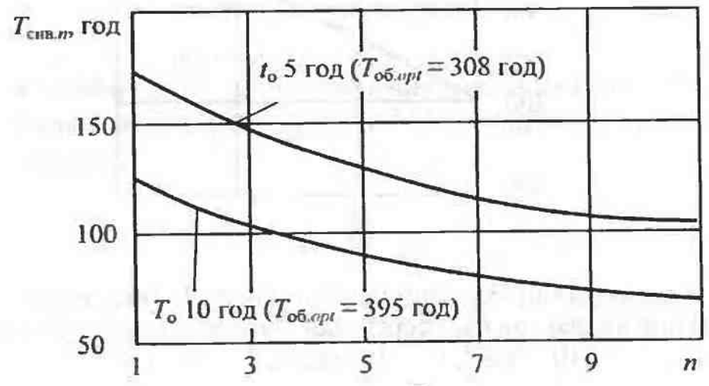
при $k = 1$, $\Delta = \sqrt{n}$.

Якщо швидкість зміни параметра поступових відмов різна в проміжках між циклами ТО, то

$$\Delta = \sqrt{1 + \frac{2}{nb}\sum_{m=1}^{n}\sum_{j=1}^{m=1} b_j}.$$



а



б

Рис. 2.5. Залежності оптимальної періодичності ТО (*а*) та середнього наробітку на відмову між сусідніми циклами ТО (*б*) від тривалості експлуатації (кількості циклів ТО) при $k = 0{,}8$, $t_в = 1$ год, $a = 10^{-3}$ год$^{-1}$, $b = 10^{-5}$ год$^{-2}$, $c = 10^{-7}$ год$^{-3}$

Унаслідок фізичного зносу і старіння елементів наробіток на відмову зі зростанням порядкового номера циклу ТО (з перебігом часу експлуатації) зменшується. У випадку забезпечення фактичного відновлення всіх елементів системи ($k = 1$) наробіток на відмову не зменшується (рис. 2.6). Однак у реальних складних технічних пристроях $k < 1$. При цьому чим менша глибина ТО, тим менше середнє значення наробітку на відмову між циклами ТО.

Збільшення до розумних меж кількості циклів ТО на фіксованому відрізку експлуатації $t_р$ дає змогу збільшувати середнє значення наробітку на відмову, причому, чим більше глибина ТО, тим більше середній наробіток на відмову (рис. 2.7).

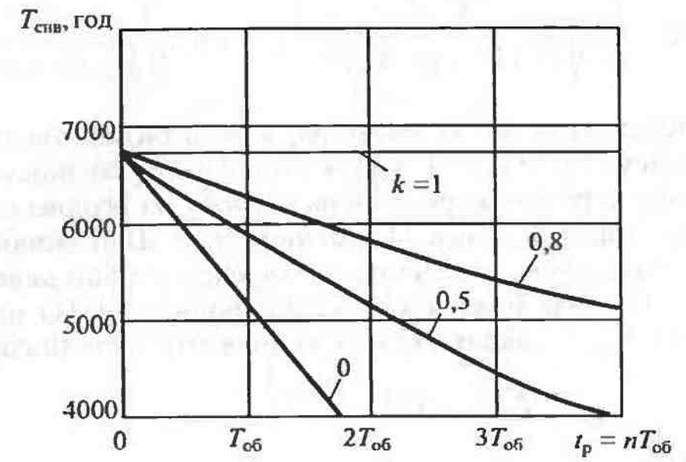


Рис. 2.6. Залежності середнього наробітку на відмови від тривалості експлуатації при $T_{об} = 10^3$ год, $a = 10^4$ год$^{-1}$, $b = 10^7$ год$^{-2}$

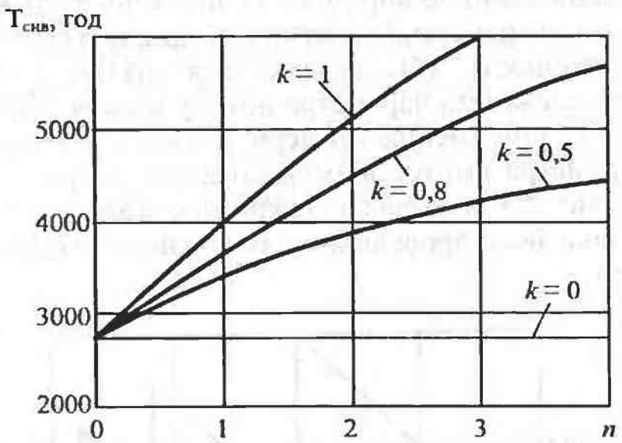


Рис. 2.7. Залежності середнього наробітку на відмови від кількості циклів ТО на відрізку часу експлуатації $t_р = 5000$ год, $a = 10^4$ год$^{-1}$, $b = 10^7$ год$^{-2}$

З використанням виразу (2.5) можна визначити середній наробіток системи на відмову з урахуванням проведеного ТО за наробітку:

$$T_{снв.сер} = \left(a + \frac{1}{2} bT_{об} + \frac{1}{2}(1-k)bnT_{об} \right)^{-1}. \qquad (2.8)$$

При визначенні середнього наробітку на відмову $T_{снв.сер}$ за оптимальної періодичності ТО і заданої тривалості експлуатації системи $nT_{об}$ треба врахувати, що $T_{об} = T_{об.орт}$. Тоді

$$T_{\text{снв.сер}} = \left(a + \frac{1}{2} b \sqrt{\frac{2t_{\text{о}}}{bt_{\text{в}}\left(k + n(1-k)\right)}} + \frac{1}{2}(1-k)bn \sqrt{\frac{2t_{\text{о}}}{bt_{\text{в}}\left(k + n(1-k)\right)}} \right)^{-1}.$$

Якщо періодичність ТО постійна, навіть оптимізована за будь-яким критерієм, то при $k < 1$ настає етап фізичного зносу обладнання. При цьому середній наробіток на відмову на інтервалах експлуатації між сусідніми циклами ТО зменшується. Щоб запобігти фізичному зносу апаратури, необхідно під час експлуатації дедалі частіше проводити ТО. Тоді зв'язок між періодичностями ТО на першому $T_{\text{об.}opt_1}$ та i-му $T_{\text{об.}opt_i}$ циклах ТО можна виразити в такий спосіб:

$$T_{\text{об.}opt_i} = (2k-1)^{i-1} T_{\text{об.}opt_1}, \qquad (2.9)$$

де $k \geq 0{,}5$; $i \geq 1$.

Дотримання співвідношення (2.9) дає змогу забезпечити сталість середнього наробітку системи на відмови. Тут значення $T_{\text{об.}opt_1}$ визначається відповідно до виразу (2.7), після чого для кожного наступного відрізка експлуатації (наступного циклу ТО) оптимальне значення періодичності ТО визначається згідно з виразом (2.9). Розрахована залежність параметра потоку відмов обладнання від часу експлуатації при зменшенні періодичності ТО задля збереження сталості параметра потоку відмов показана на рис. 2.8. Для цього випадку на рис. 2.9 показана залежність оптимальної періодичності ТО від глибини його проведення для кожного i-го відрізка експлуатації системи.

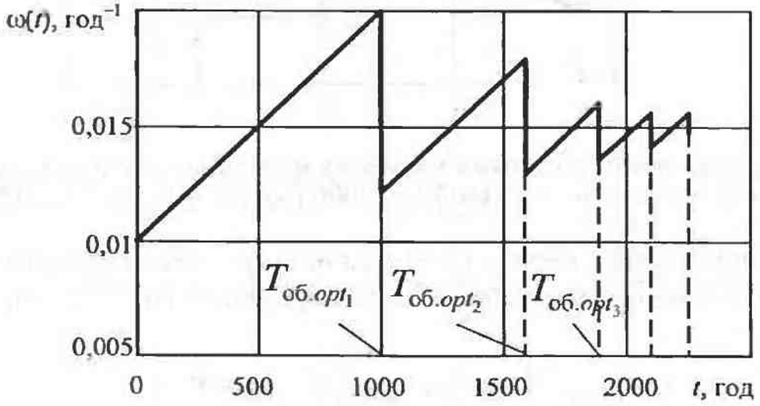


Рис. 2.8. Залежність параметра потоку відмов системи, що обслуговується, від часу експлуатації при $T_{\text{об.}opt_1} = 10^3$ год, $a = 10^{-2}$ год$^{-1}$, $b = 10^{-4}$ год$^{-2}$, $k = 0{,}8$

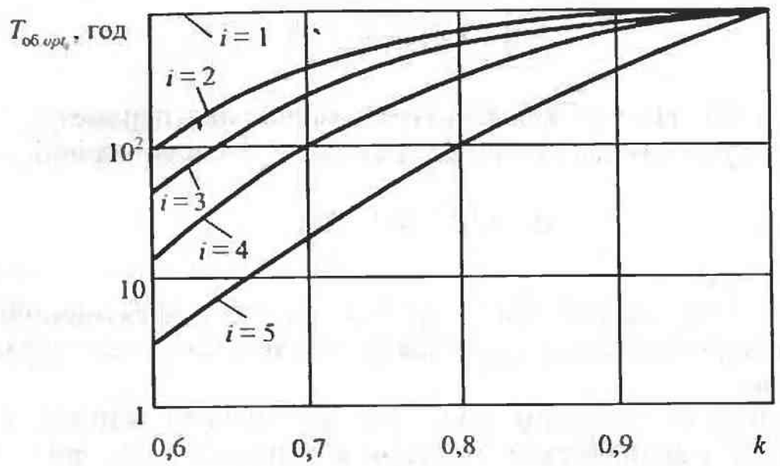


Рис. 2.9. Залежності оптимальної періодичності ТО від глибини його проведення при $T_{\text{об.}opt_1} = 10^3$ год, $T_{\text{снв}} = \text{const}$

Тільки для першого відрізка експлуатації (до проведення першого циклу ТО) оптимальна періодичність не залежить від глибини ТО.

Періодичність ТО за станом визначається станом контрольованих параметрів обладнання. Технічне обслуговування за станом дає змогу збільшити ресурс обладнання, що обслуговується $t'_{\text{р}2}$, порівняно з ресурсом, що не обслуговується $t'_{\text{р}1}$, в Δ_1 раз:

$$\Delta_1 = \frac{t'_{\text{р}2}}{t'_{\text{р}1}} = \sqrt{\alpha},$$

де α — коефіцієнт, що показує, у скільки разів ТО за станом зменшує параметр потоку відмов поступового характеру ($\alpha = 1$).

Якщо замість ТО, регламентованого за наробітком, проводити ТО за станом, то збільшення ресурсу

$$\Delta_2 = \frac{t'_{\text{р}2}}{t'_{\text{р}1}} = \sqrt{\frac{\alpha}{n}\left(k + n(1-k)\right)}.$$

При $k = 1$ $\Delta_2 = \sqrt{\frac{\alpha}{n}}$.

Технічне обслуговування за станом сприяє збільшенню ресурсу обладнання порівняно з ТО за наробітком, якщо дотримується нерівність

$$\frac{\alpha}{n}\left(k+n(1-k)\right)>1.$$

При цьому ТО має забезпечувати зменшення параметра потоку відмов поступового характеру в α раз, що зумовлено нерівністю

$$\alpha > n\left(k+n(1-k)\right)^{-1}.$$

При $k=1$ $\alpha > n$.

Обслуговування системи за станом тоді буде ефективнішим, коли більш повно здійснюватиметься контроль зміни в часі параметрів обладнання.

У процесі експлуатації необхідно забезпечити мінімальну вартість витрат з використання системи в одиницю часу, що визначається виразом

$$C=(1-K_{тв})C_{зб}+C_{в}+\frac{1}{T_{об}}C_{ТО}, \qquad (2.10)$$

де $C_{зб}$ — збиток від простою системи в одиницю часу; $C_{в}$ — вартість технічної експлуатації системи в одиницю часу; $C_{ТО}$ — витрати на проведення одного циклу ТО.

Підставивши у вираз (2.10) значення $K_{тв}$ з формули (2.4) при $\omega(t)=\alpha+bt$, $k=1$, одержимо вираз, дослідження на екстремум якого дає змогу дістати оптимальну періодичність ТО за мінімальними витратами:

$$T_{об.opt}=\sqrt{\frac{2}{bt_{в}}\left(t_{о}+\frac{C_{ТО}}{C_{зб}}\right)}.$$

При $C_{ТО} >> C_{зб}$ одержимо вираз (2.7).

2.2.2.4. Періодичність технічного обслуговування обладнання КС та М при збереженні

Умови збереження апаратури впливають на характеристики її надійності. Параметр відмов апаратури і коефіцієнт простою, під яким розуміється ймовірність того, що в будь-який момент апаратура, що зберігається, виявиться непрацездатною, за несприятливих умов збереження буде вище.

Позначимо тривалість циклу збереження апаратури між проведенням перевірок через $T_{цз}$, тоді $T_{цз}=t_{г}+t_{нг}$, де $t_{г}$ — середня тривалість перебування апаратури в стані готовності в межах будь-якого

циклу збереження; $t_{нг}$ — середня тривалість перебування апаратури в стані неготовності в межах циклу збереження.

Для зменшення коефіцієнта простою необхідно зменшити тривалість перебування апаратури протягом циклу збереження $T_{цз}$ в стані неготовності $t_{нг}$, що, своєю чергою, дорівнює:

$$t_{нг}=t_{нс}+t_{о}.$$

Тут $t_{нс}$ — середня тривалість перебування апаратури в непрацездатному стані між двома перевірками; $t_{о}$ — середня тривалість однієї операції контролю і ТО, що включає усунення ушкоджень, що виконуються в процесі перевірки.

У період між перевірками апаратура може виявитися в непрацездатному стані як унаслідок відмов, що виникають а процесі збереження між перевірками, так і внаслідок відмов, що виникають при її вимиканні після попередньої перевірки:

$$T_{нс}=t_{нс1}+t_{нс2}.$$

Припустимо, що всі перевірки апаратури однотипні, період їх проведення детермінований, а глибина контролю і ТО при перевірках забезпечує виявлення й усунення всіх наявних ушкоджень.

Середній час перебування апаратури в непрацездатному стані за рахунок відмов між сусідніми циклами перевірки і ТО можна визначити за формулою

$$t_{нс1}=\int_{0}^{T_{цз}}(T_{цз}-t)f(t)dt, \qquad (2.11)$$

де $f(t)$ — щільність розподілу тривалості часу безвідмовної роботи при збереженні апаратури.

При експонентному законі розподілу тривалість часу безвідмовної роботи при збереженні

$$f(t)=\lambda_{зб}e^{-\lambda_{сер}}, \qquad (2.12)$$

де $\lambda_{зб}=\frac{1}{T_{зб}}$ — параметр потоку відмов апаратури при збереженні; $T_{зб}$ — середня тривалість часу безвідмовної роботи апаратури при збереженні.

Підставляючи вираз (2.12) у формулу (2.11) й інтегруючи його, одержуємо

$$t_{нс1}=T_{цз}-\frac{1}{\lambda_{зб}}\left(1-e^{-\lambda_{зб}T_{цз}}\right). \qquad (2.13)$$

Обмежуючи трьома першими членами розкладання показової функції в ряд

$$e^{-\lambda_{зб}t} = 1 - \lambda_{зб}T_{цз} + \frac{1}{2}\left(\lambda_{зб}T_{цз}\right)^2,$$

перетворимо вираз (2.13) до вигляду

$$t_{нс1} \approx \frac{1}{2}\lambda_{зб}T_{цз}^2.$$

Оскільки відмови, що виникають при вмиканні апаратури після перевірки, виявляються й усуваються тільки в ході наступної перевірки, то середня тривалість перебування апаратури в непрацездатному стані з цієї причини

$$t_{нс2} = q_{сер}\left(T_{цз} - t_{о}\right) \approx q_{сер}T_{цз},$$

де $q_{сер}$ — середня ймовірність того, що апаратура відмовить при вимиканні після перевірки. З огляду на те, що за весь термін збереження проводиться n перевірок, імовірність

$$q_{сер} = \frac{1}{n}\sum_{i=1}^{n} iq_1, \qquad (2.14)$$

де q_1 — імовірність відмови апаратури після першої перевірки. При заданому ресурсі збереження $t_{зб}$ кількість циклів контролю

$$n = \frac{t_{зб}}{T_{цз}}. \qquad (2.15)$$

Оскільки $\sum_{i=1}^{n} i = \frac{1}{2}n(n+1)$, співвідношення (2.14) набере вигляду

$$q_{сер} = \frac{1}{2}(n+1)q. \qquad (2.16)$$

З урахуванням виразів (2.15) і (2.16) матимемо, що найшкідливіший час перебування апаратури в непрацездатному стані через відмови при її вимиканні дорівнює

$$T_{нс2} = \frac{1}{2}\left(\frac{t_{зб}}{T_{цз}} + 1\right)T_{цз}q_1.$$

Загальний час неготовності апаратури до використання при збереженні за період між послідовними циклами контролю

$$t_{нг} = \frac{1}{2}\lambda_{сер}T_{цз}^2 + \frac{1}{2}\left(\frac{t_{зб}}{T_{цз}} + 1\right)T_{цз}q_1 + t_{о}.$$

З огляду на те, що коефіцієнт простою

$$K_{п} = \frac{t_{нг}}{T_{цз} + t_{нг}} \cong \frac{t_{нг}}{T_{цз}},$$

одержуємо вираз:

$$K_{п} = \frac{1}{2}\lambda_{зб}T_{цз} + \frac{1}{2}\left(\frac{t_{зб}}{T_{цз}} + 1\right)q + \frac{t_{о}}{T_{цз}}. \qquad (2.17)$$

Якщо періодичність контролю вибирається з умови забезпечення мінімуму коефіцієнта простою апаратури, то для одержання оптимальної періодичності контролю при збереженні треба диференціювати вираз (2.17) за періодичністю контролю $T_{цз}$ й отримане співвідношення дорівняти нулю:

$$\frac{1}{2}\lambda_{зб} - \frac{1}{2}\frac{t_{зб}q_1}{2T_{зб}^2} - \frac{t_{о}}{T_{зб}^2} = 0. \qquad (2.18)$$

З рівняння (2.18) дістаємо, що оптимальна періодичність контролю апаратури при збереженні

$$T_{зб} = \sqrt{\frac{q_1 t_{зб} + 2t_{о}}{\lambda_{зб}}}.$$

Якщо при контролі стану апаратури, що зберігається, у разі вмикання відмов не виникає або апаратура взагалі не вмикається, то оптимальна періодичність контролю

$$T_{цз.opt} = \sqrt{\frac{2t_{о}}{\lambda_{зб}}}.$$

Однак практично при перевірці вдається знайти й усунути не усі виникаючі ушкодження, що ускладнює розрахунок значення $T_{цз.opt}$.

У процесі збереження обладнання можуть відбуватися його відмови, що мають раптовий або поступовий характер. За лінійного закону зміни параметра потоку відмов $\omega_{зб}(t) = a_{зб} + b_{зб}t$ імовірність безвідмовного збереження апаратури

$$P_{зб}(t) = \exp\left(-\int_0^t (a_{зб} + b_{зб}u)\,du\right).$$

Щільність імовірності відмов при цьому виражається так:

$$f(t) = P'_{зб} = (a_{зб} + b_{зб}u)\exp\left(-\left(\alpha_{зб}t + \frac{1}{2}b_{зб}t^2\right)\right).$$

При періодичному контролі стану апаратури середній час її перебування в несправному стані

$$t_{сер.нс} = \int_0^{T_{цз}} (T_{цз} - t) f(t)dt = \int_0^{T_{цз}} (T_{цз} - t)(a + bt)e^{-\left(a_{зб}t + \frac{1}{2}b_{зб}t^2\right)}dt.$$

За період між послідовно проведеними перевірками середній час відновлення дорівнює добуткові середнього числа виниклих відмов $(a_{зб} + b_{зб}t)T_{цз}$ на середній час відновлення наслідку однієї відмови $t_в$:

$$t_{в\Sigma} = (a_{зб} + b_{зб}t)T_{цз}t_в,$$

а загальний час простою апаратури

$$t_{п\Sigma} = (a_{зб} + b_{зб}t)T_{цз}t_в + t_о + \int_0^{T_{цз}} (T_{цз} - t)(a_{зб} + b_{зб}t)e^{-\left(a_{зб}t + \frac{1}{2}b_{зб}t^2\right)}dt.$$

Коефіцієнт простою при цьому виражається так:

$$K_п = \frac{t_{п\Sigma}}{T_{цз} + t_{п\Sigma}} \approx \frac{t_{п\Sigma}}{T_{цз}} = \frac{t_о}{T_{цз}}\left(a_{зб} + b_{зб}T_{цз}\right)t_в + \int_0^{T_{цз}} (T_{цз} - t)(a_{зб} + b_{зб}t)e^{-\left(a_{зб}t + \frac{1}{2}b_{зб}t^2\right)}dt. \quad (2.19)$$

Знайшовши похідну виразу (2.19) за $T_{цз}$ і дорівнявши її до нуля, можна визначити оптимальну періодичність контролю з виразу:

$$b_{зб}t_в T_{цз}^2 + \frac{d}{dT_{цз}}\left(\int_0^{T_{цз}} (T_{цз} - t)(\alpha_{зб} + b_{зб}t)e^{-(\alpha_{зб}t + \frac{1}{2}b_{зб}t^2)}dt\right)T_{цз} - \int_0^{T_{цз}} (T_{цз} - t)(a_{зб} + b_{зб}t)e^{-(\alpha_{зб}t + \frac{1}{2}b_{зб}t^2)}dt - t_о = 0.$$

Рішення цього рівняння здійснюється числовими методами на ЕОМ.

Методичні вказівки

Будь-яке технічне обладнання КС та М в процесі експлуатації вимагає визначеного ТО. При розв'язанні завдання організації ТО (вибір стратегії, періодичності й обсягу) необхідно попередньо вивчити принцип дії системи, склад обладнання та його взаємодію, конструктивні особливості, звернути увагу на режими його роботи, наявність апаратурного, функціонального і часового резервів, умови роботи, кількість і кваліфікацію технічного персоналу.

Вибираючи стратегію ТО необхідно враховувати можливості забезпечення необхідного рівня надійності.

Питання для самоперевірки

1. Що являє собою ТО КС та М?
2. Перелічіть використовувані стратегії ТО.
3. У чому суть регламентованого ТО?
4. Як здійснюється ТО за станом?
5. Як враховується стан технічних і вихідних тактичних характеристик радіоелектронних систем при організації ТО?
6. З огляду на які міркування вибираються періодичність й обсяг ТО?
7. Наведіть приклад, коли періодичність ТО може бути оптимізована.
8. Що розуміється під глибиною ТО апаратури?
9. Як впливає ТО на довговічність апаратури?
10. Як змінюється середній наробіток на відмови апаратури між двома циклами ТО в процесі експлуатації?
11. У чому особливості ТО апаратури при її збереженні?
12. Як визначається оптимальна періодичність ТО при збереженні апаратури?

2.2.3. Організація ремонтного обслуговування КС та М

2.2.3.1. Види ремонту

Ремонт — це комплекс заходів (операцій) з відновлення справності або працездатності виробів і відновлення їх ресурсів. Відновлення ресурсів проводять при плановому ремонті, а відновлення працездатності виробів — при позаплановому.

Розрізняють поточний, середній і капітальний ремонт. Зазвичай в умовах експлуатації роблять поточний і в ряді випадків — середній ремонт. Капітальний ремонт роблять на спеціальних ремонтних підприємствах.

Поточний ремонт може бути плановим і позаплановим. Позаплановий ремонт здійснюється для усунення наслідку поступових і раптових відмов апаратури.

Плановий ремонт може бути регламентованим і здійснюватися за технічним станом.

Регламентований ремонт — це плановий ремонт, виконуваний з періодичністю й в обсязі, установленими нормативно-технічною документацією, незалежно від технічного стану виробу в момент початку ремонту. Періодичність його регламентують за наробітком апаратури (ресурсний принцип) і за календарними або сезонними термінами. Терміни проведення планового ремонту встановлюють заздалегідь. Метою цього ремонту є заміна елементів апаратури, що стали непридатними або невідповідними установленим нормам.

Ремонт за технічним станом — це плановий ремонт, якому передує контроль технічного стану апаратури, що дає змогу зробити висновок про необхідність проведення ремонту. При цьому контроль технічного стану, що може бути безперервним або періодичним, виконується з періодичністю й в обсязі, установленими нормативно-технічною документацією. Обсяг і момент початку такого виду ремонту визначається технічним станом апаратури.

Середній і капітальний ремонт — планові заходи. Основне їх завдання полягає в усуненні всіх ушкоджень, що накопичувалися, і відновленні ресурсу апаратури задля забезпечення необхідного рівня її надійності до чергового планового ремонту подібного рівня.

Необхідність чергування двох видів планового ремонту (середнього і капітального) зумовлена тим, що комплектуючі елементи в деталі не є однаково надійними й вимагають різних витрат часу, спеціальних механізмів і пристроїв для їх заміни. Звичайно між двома послідовно проведеними капітальними ремонтами роблять ряд середніх ремонтів, а між двома середніми — ряд ремонтів регламентованого характеру або за технічним станом. Терміни проведення позапланового поточного ремонту носять випадковий характер.

Перед проведенням планових видів ремонту виконують такі підготовчі роботи: складають дефектну відомість, що заповнюється в процесі ретельного аналізу стану апаратури; визначають обсяг ремонтних робіт; складають і затверджують кошториси; забезпечують фінансування робіт; складають план організації робіт і ін.

У ремонт апаратуру передають ремонтному підприємству за актом, у якому відбивають її технічний стан і комплектність. З актом передають паспорт і формуляр. По закінченні ремонту всі роботи за актом приймає спеціальна комісія, призначувана керівниками експлуатаційного підприємства і ремонтної організації.

Ремонтне підприємство має випускати апаратуру після ремонту справною і гарантувати її працездатність протягом визначених термінів.

Трудомісткість ремонту визначається трудовитратами на проведення одного виду ремонту виробу і виражається в людиногодинах. Трудовитрати залежать від кваліфікації персоналу, прийнятої системи та технології ремонту, системи матеріально-технічного забезпечення, оснащеності спеціальним обладнанням, приладами, інструментом. Тож величина, що характеризує трудомісткість, стає випадковою. Тому найбільш повними характеристиками ремонту є його середня сумарна і питома сумарна трудомісткості.

Середня сумарна трудомісткість ремонтів $\theta_{\Sigma ср}$ являє собою математичне сподівання сумарних трудовитрат θ_{Σ} на всі види ремонту апаратури за визначений період експлуатації:

$$\theta_{\Sigma ср} = M(\theta_{ср}) = \int_0^\infty \theta_\Sigma f(\theta_\Sigma) d\theta_\Sigma,$$

де $f(\theta_\Sigma)$ — щільність імовірності сумарних трудовитрат.

Питома сумарна трудомісткість ремонтів γ_0 визначається як відношення середньої сумарної трудомісткості ремонтів до сумарного наробітку об'єктів за той самий період експлуатації.

Економічна сторона процесу ремонту апаратури може характеризуватися вартістю ремонту даної апаратури при усуненні чергової несправності або сумарною вартістю проведення усіх видів її ремонту за визначений період експлуатації. До вартості входять амортизаційні витрати контрольно-вимірювальної апаратури, інструменту, витрати електроенергії тощо.

Середня сумарна вартість ремонту $C_{\Sigma ср}$ визначається як математичне сподівання сумарних витрат на усі види ремонтів за визначений період експлуатації:

$$C_{\Sigma ср} = M(C_\Sigma).$$

Питома сумарна вартість ремонту — це відношення середньої сумарної вартості ремонту до середнього значення сумарного наробітку об'єкта за той самий період експлуатації.

2.2.3.2. Методи ремонту

Методи ремонту класифікуються за ознакою належності ремонтованих елементів відповідно до еталону його виконання, потоку виробництва, закріплення об'єкта за виконавцями.

За ознакою належності ремонтованих елементів розрізняють такі методи ремонту:

— незнеособлений метод ремонту, що передбачає збереження належності відновлюваних елементів до визначеного виробу. У процесі ремонту на даний виріб установлюють усі належні йому справні елементи (блоки, плати, агрегати). Цей метод забезпечує високу продуктивність праці, скорочує тривалість ремонту і фронт робіт;

— знеособлений метод ремонту, що не передбачає збереження належності елементів до визначеного виробу. Елементи, зняті з ремонтованих виробів, після їх перевірки і відновлення можуть бути встановлені на будь-який виріб, що є в ремонті. Цей метод найефективніший. Він сприяє потоковій організації робіт, автоматизації і механізації процесу ремонту. Однак після його завершення потрібне додаткове регулювання і настроювання апаратури;

— мішаний метод — проміжний між двома попередніми, при цьому методі ремонту не знеособлюються тільки найбільш відповідальні вузли й агрегати виробу;

— агрегатний метод, який полягає в заміні укрупнених агрегатів виробу, що відмовили, справними, котрі на об'єкті не ремонтуються. Агрегати, що відмовили, надходять на ремонтні підприємства для централізованого ремонту. Це значно скорочує час простою обладнання на відновленні, полегшує його експлуатацію, оскільки не потрібно виявлення елемента, що відмовив, та його заміна на об'єкті.

Використання того або іншого методу ремонту визначається особливостями конструкції системи й організацією ремонтної служби.

Залежно від етапності проведення розрізняють такі методи ремонту:

- поетапний метод, за якого весь обсяг ремонтних робіт поділяється на частини і проводиться поетапно;
- поетапно-блоковий метод, який передбачає відновлення на кожному з етапів тільки визначених блоків, агрегатів апаратури, що сприяє скороченню часу простою обладнання;
- метод безперервного ремонту, при якому елементи, вузли, блоки, агрегати відновлюються в процесі оперативного технічного обслуговування малої періодичності.

Залежно від потоковості виробництва розрізняють стендовий метод, за якого всі необхідні операції з даним виробом виконують на

спеціальному стенді, і потоковий метод, за якого на кожному зі стендів виконують тільки визначені операції ремонту.

Залежно від закріплення за виконавцями розрізняють індивідуальний і бригадний методи ремонту, а також метод ремонту спеціалізованою організацією.

В КС та М ремонтопридатність апаратури та її висока надійність забезпечуються автоматизацією пошуку й локалізацією виникаючих ушкоджень, оперативною заміною несправної плати або блоку після їх відмови, використанням централізованого ремонту, настроюванням та перевіркою замінених плат і блоків у централізованій ремонтній майстерні.

2.2.3.3. Поновлювальність апаратури

Надійність апаратури тривалого та багаторазового використання характеризується не тільки безвідмовністю, а й поновлювальністю, що охоплює досить широке поняття, зумовлене як властивостями самої апаратури, так і організацією процесу її експлуатації.

Поновлювальність залежить від таких чинників:

- конструктивної пристосованості апаратури до виявлення й усунення ушкоджень;
- рівня кваліфікації обслуговуючого персоналу;
- відповідальності обслуговуючого персоналу за доручену справу;
- організації процесу обслуговування, створення необхідних умов роботи, забезпечення необхідними запасними елементами.

Під ремонтопридатністю розуміють властивість апаратури до запобігання, виявлення й усуненню ушкоджень.

Поновлювальність є однією з важливих сторін надійності апаратури.

Загалом час ремонту можна поділити на три складові:

✓ час установлення факту наявності відмови або ушкодження;

✓ технічний час ремонту, куди входять підготовка апаратури до ремонту, час підготовки контрольно-вимірювальної апаратури й інструменту, час пошуку відмови (її локалізації), час її усунення, а також час, необхідний на післяремонтне обслуговування і регулювання;

✓ непродуктивні втрати часу організаційного характеру (простої апаратури в очікуванні відновлення) у зв'язку з відсутністю запасних елементів і приладів (ЗІП) або доставкою обслуговуючого персоналу.

Дуже часто співвідношення між технічним часом відновлення й непродуктивними втратами часу складаються не на користь технічного часу. Удосконалюючи організацію ремонту, необхідно скорочувати і технічний час ремонту, що залежить від кваліфікації техніч-

ного персоналу, його відповідальності за роботу, наявності необхідних умов, ЗІП й інструменту.

Час відновлення апаратури багато в чому визначається її ремонтопридатністю, що залежить від її конструкції, наявної системи контролю працездатності. Тому при розробці апаратури можна значною мірою впливати на характеристики її поновлювальності, а отже, й надійності.

Усі зазначені складові ремонту є спільними для всіх методів ремонту, незалежно від методу виявлення ушкодження — автоматичного або ручного.

При ручному пошуку елемента, що відмовив, і ремонті методом його заміни в апаратурі блокової конструкції установлення факту ушкодження становить в середньому 3 %, установлення характеру ушкодження і пошук елемента, що відмовив, — 61 %, усунення наслідку ушкодження — 15 %, перевірка апаратури після ремонту і її регулювання — 21 % від загального часу ремонту. Застосування систем діагностування елемента, що відмовив, може істотно перерозподіляти час, затрачуваний на ремонт.

Поновлювальність апаратури характеризується середнім часом її відновлення $t_в$, законом розподілу часу відновлення, ймовірністю відновлення.

Якщо $\nu(t)$ — параметр закону розподілу часу відновлення, то ймовірність своєчасного відновлення визначається так:

$$V(t)=1-\exp\left(\int_0^t \nu(u)du\right).$$

При експонентному законі розподілу:

$$\nu(t)=\mu,\ f(t)=\mu e^{-\mu t},\ V(t)=1-e^{-\mu t},$$

де $\mu=\dfrac{1}{t_в}$ — інтенсивність відновлення апаратури; $t_в$ — середній час відновлення; $f(t)$ — щільність імовірності своєчасного відновлення.

Закон розподілу активного часу ремонту в основному визначається методом пошуку ушкодження і конструкцією апаратури. Для апаратури модульної конструкції, де ремонт здійснюється заміною модуля, часто має місце експонентний закон розподілу часу ремонту. Він справедливий і для відносно простої апаратури.

Найбільш яскраво експонентний закон розподілу часу відновлення виявляється при ремонті ЕОМ, пошук відмов у яких здійснюється за допомогою тестового контролю.

Для тих систем, при відновленні яких потрібні значні витрати ручної праці, закон розподілу часу відновлення не є експонентним. Якщо для них характерні порівняно мала кількість відновлень за малі і великі проміжки часу t та досить часті випадки відновлення за час $t - t_в$, то закон розподілу часу відновлення в цьому випадку добре описується розподілом Ерланга другого порядку:

$$f(t)=\frac{4t}{t_в^2}\exp\left(-\frac{2t}{t_в}\right).$$

Часто зустрічається логарифмічно-нормальний розподіл часу відновлення

$$f(t)=\frac{0{,}443}{\sigma t\sqrt{2\pi}}\exp\left(-\frac{\lg t-\lg t_в}{2\sigma^2}\right),$$

де σ — середньоквадратичне відхилення $\lg t$ від середнього значення $\lg t_в$.

При логарифмічно-нормальному розподілі ймовірність відновлення

$$V(t)=F_0\left(\frac{\lg t-\lg t_в}{\sigma}\right),$$

де $F_0(x)$ — табульована функція вигляду

$$F_0(x)=\int_{-\infty}^{x}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(x-a)}{2\sigma^2}\right)dx.$$

Математичне сподівання $t_в$ і найвірогідніше значення часу відновлення $t_м$ визначається в такий спосіб:

$$\lg t_в=\lg t_м+1{,}15\sigma^2;$$

$$\lg t_м=\lg t_в-2{,}3\sigma^2.$$

Для закону Ерланга другого порядку можна одержати ймовірність відновлення в такий спосіб:

$$V(t)=1-\left(1+\frac{2t}{t_в}\right)\exp\left(\frac{2t}{t_в}\right).$$

Порівнюючи експонентний розподіл і розподіл Ерланга другого порядку (рис. 2.10), можна дійти висновку, що залежності $V(t)$ за однакового середнього значення часу відновлення $t_в$ помітно відрізняються тільки при $t < t_в$.

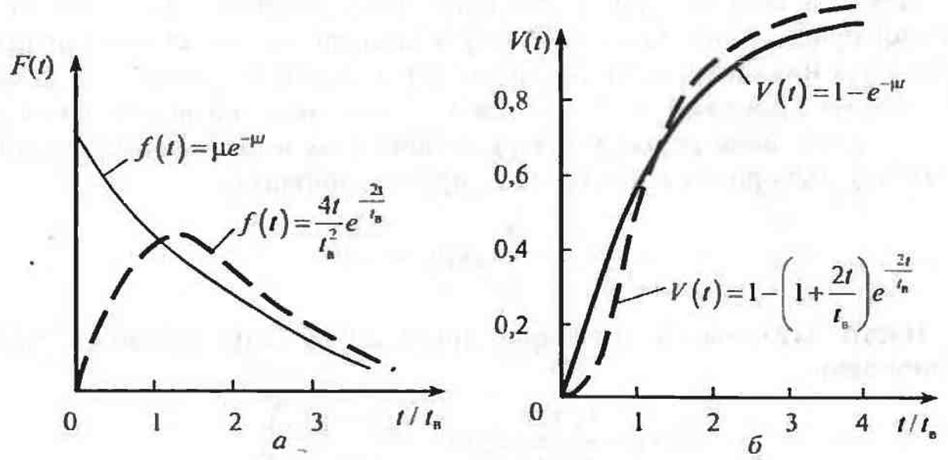


Рис. 2.10. Закони розподілу часу відновлення (*а*)
(експоненгний та Ерланга другого порядку) і ймовірність відновлення (*б*)
залежно від відношення часу відновлення та його середнього значення

Якщо ж $t \geq t_в$, то криві настільки близькі, що при інженерних розрахунках поновлювальності можна користуватися будь-яким із цих законів.

2.2.3.4. Методи пошуку ушкоджень

Виявлення несправності і її локалізація вимагають найбільших витрат часу (до 70 %) при ремонті апаратури. Розробка заходів для спрощення пошуку ушкоджень скорочує не тільки частку активного часу ремонту, а й зменшує фізичні і розумові навантаження обслуговуючого персоналу. Виникнення ушкоджень і відмов систем відповідального призначення викликає значні психологічні навантаження. Тому необхідна автоматизація процесу пошуку ушкоджень. Однак вона має здійснюватися тільки тоді, коли витрати на створення й експлуатацію надійних і дорогих систем виправдуються підвищенням ремонтопридатності апаратури.

При пошуку ушкоджень під елементом прийнято розуміти таку самостійну конструктивну частину апаратури, що у випадку наявності в ній несправності цілком заміняється.

Процес пошуку несправного елемента має дві стадії:

— вибір послідовності перевірок;

— вибір методики проведення окремих операцій перевірки.

Проаналізуємо послідовність перевірок системи, яка складається з n елементів, що забезпечує найменший час пошуку ушкодження. Нехай відмови елементів незалежні, а відмови кожного з них при-

зводить до відмови даного обладнання. Звичайно відомі інтенсивності відмов елементів і час перевірки їхньої справності. Середній час пошуку ушкодження в схемі

$$t_{\text{пу}} = \sum_{i=1}^{n=1} g_i^{t_{\text{пу}.i}}, \tag{2.20}$$

де g_i — умовна ймовірність відмови i-го елемента за умови відмови обладнання; $t_{\text{пу}.i}$ — середній час, затрачуваний на пошук ушкодженого i-го елемента.

Своєю чергою, час пошуку i-го ушкодженого елемента являє собою суму витрат часу на перевірку j елементів схеми:

$$t_{\text{пу}} = \sum_{j=1}^{i} \tau_j, \tag{2.21}$$

де τ_j — середній час перевірки справності j-го елемента. Умовна ймовірність ушкодження j-го елемента g_i дорівнює відношенню інтенсивності відмов i-го елемента λ_i до інтенсивності відмов усієї схеми з n послідовно з'єднаних елементів:

$$g_i = \frac{\lambda_i}{\sum_{j=1}^{n} \lambda_i}.$$

Тоді з урахуванням виразів (2.20) і (2.21) дістаємо

$$t_{\text{пу}} = \sum_{i=1}^{n=1} g_i \sum_{j=1}^{i} \tau_j.$$

Нехай існує можливість подати на вхід першого елемента схеми з n послідовно з'єднаних елементів контрольний сигнал і виміряти реакцію кожного елемента на цей сигнал. Відмова може бути в кожному з n елементів схеми. Отже, ентропія (невизначеність) стану така:

$$H = -\sum_{i=1}^{m} q_i \log_2 q_i.$$

Необхідно розв'язати завдання, в якій частині схеми міститься несправний елемент. Для цього вимірюється реакція на вхідний сигнал в одній із точок схеми. Якщо реакція відповідна необхідній, то робиться висновок, що ушкоджений елемент міститься після обраної контрольної точки схеми. Якщо ж реакція не відповідає необхід-

ній, то несправний елемент міститься між входом і контрольною точкою схеми.

Після першого вимірювання одержують таку кількість інформації:

$$I = -\sum_{i=1}^{k} q_1 \log_2 \sum_{i=1}^{k} q_1 - \left(1 - \sum_{i=k+1}^{k} q_1 \log_2 1 - \sum_{i=k+1}^{n} q_i\right), \qquad (2.22)$$

де k — номер елемента, за яким вимірюється реакції на вхідний сигнал (контрольний). Якщо позначити

$$\sum_{i=1}^{k} q_i = P_k,$$

то вираз (2.22) набере такого вигляду:

$$I = -P_k \log_2 P_k - (1 - P_k)\log_2(1 - P_k).$$

Скорочення часу пошуку ушкодження може бути досягнуте зменшенням кількості вимірювань. Однак для цього необхідно збільшити кількість інформації, одержуваної при кожному вимірюванні.

Для визначення умови отримання найбільшої кількості інформації при вимірюваннях дорівняємо похідну $\frac{dI}{P_k}$ нулю. З рішення здобутого рівняння маємо:

$$\log_2 P_k = \log_2(1 - P_k).$$

Звідси $P_k = I - P_k$; $P_k = 0{,}5$.

Те саме рівняння можна записати по-іншому:

$$\sum_{i=1}^{k} q_i = \sum_{i=1}^{k} \lambda_i \left(\sum_{i=1}^{n} \lambda_i\right)^{-1} = 0{,}5.$$

Таким чином, для досягнення оптимальної процедури пошуку ушкодження необхідно кожного разу проводити вимірювання реакції на контрольний сигнал в точці схеми, яка ділить її навпіл за ймовірністю (інтенсивністю) відмов. Цей метод пошуку ушкодження отримав назву методу половинного ділення.

Використовуючи метод половинного ділення, алгоритм пошуку ушкодження схеми будується таким чином:

— схема розділяється за умовною ймовірністю відмови навпіл і в точці ділення проводиться вимірювання реакції на вхідний сигнал;

— залежно від результату випробування визначається несправна частина схеми;

— процедура повторюється для несправної частини схеми;

— процедура повторюється до локалізації несправного елемента.

Окрім розглянутого методу, використовують методи структурно-логічного аналізу, зокрема, метод діагностичних таблиць.

При пошуку ушкодження необхідно, крім вибору послідовності перевірки несправності елементів, вибрати методику перевірки справності конкретних елементів схеми. Розрізняють такі способи перевірки справності елемента:

- проміжних вимірювань;
- зовнішнього огляду;
- заміни;
- порівняння і ін.

Спосіб проміжних вимірювань припускає вимірювання параметрів елементів або схем апаратури. Якщо параметри знаходяться в полі їх допусків, то ухвалюється рішення про справність контрольованого елемента або схеми.

При зовнішньому огляді монтажу і схеми щодо зміни зовнішнього вигляду елементів, їх перегріву, іскріння, підгорання, течі часто вдасться виявити наявність ушкодження.

Спосіб заміни полягає в заміні окремих елементів, блоків, вузлів явно справними. При відновленні ознаки нормальної роботи робиться висновок про несправність замінюваного елементів.

Спосіб порівняння полягає в порівнянні режимів роботи використовуваного елемента або схеми з однотипним, але явно справним.

Вибір способу виявлення ушкодження залежить від конструкції апаратури, наявності приладів і інструментів, кваліфікації технічного персоналу.

Автоматизація процесу пошуку несправностей дає змогу значно підвищити готовність апаратури до застосування.

2.2.3.5. Особливості технічного обслуговування комп'ютерних систем

Безперервне ускладнення логічної побудови систем, підвищення ступеня інтеграції елементів, висока щільність монтажу електронного обладнання позначаються на організації ТО комп'ютерних систем.

Розробка єдиної концепції ТО вимагає системного підходу, що враховував би взаємодію організаційних, технічних, програмних і економічних сторін цього процесу. Ця концепція є невід'ємною частиною процесу розробки технічних засобів КС та М.

Сьогодні комп'ютерні системи забезпечуються комплексом апаратно-програмних засобів підтримки експлуатації цих систем, що органічно включаються в архітектуру.

Для контролю правильності роботи ЕОМ, які входять до складу комп'ютерних систем, недопущення поширення наслідків відмов або збоїв на програми і результати обчислень, необхідно використовувати в ЕОМ системи автоматичного контролю, що сприймають помилку ЕОМ практично в момент її виникнення і перешкоджають подальшому виконанню програми.

При збої в роботі ЕОМ необхідне відновлення вірогідності інформації. Для цього створюється автоматична система відновлення обчислювального процесу. Вона викликає автоматичне повторення неправильно виконаної операції і при безпомилковому результаті повторення — відновлення обчислювального процесу. Системи автоматичного контролю відновлення забезпечують високу контролепридатність ЕОМ.

Наявність системи автоматичного діагностування полегшує і прискорює пошук ушкодження, що сприяє підвищенню коефіцієнтів готовності та технічного використання апаратури, а також скороченню часу і працезатрат при перевірці працездатності ЕОМ після ремонту або в процесі ТО.

Для зменшення часу і працезатрат при профілактичному контролі до складу ЕОМ включають систему автоматичної програмно-керованої профілактики, що здійснює автоматичну зміну рівнів вторинних напруг електроживлення, частоти і форми керівних сигналів в автоматичну реєстрацію результатів профілактичних випробувань.

Застосування автоматичних апаратно-програмних систем реєстрації й обробки інформації про відмови, збої, відновлення, проведений ТО сприяє не тільки удосконалюванню процесу ТО (оптимізації періодичності й обсягу ТО), а й прийняттю обгрунтованих і своєчасних рішень щодо заміни елементів, проведення доробок апаратури.

Методичні вказівки

Комп'ютерні системи, що входять до складу систем управління технологічними процесами (наприклад, автоматизовані системи управління повітряним рухом), є системами багаторазового тривалого використання. За час їх роботи виникають тисячі ушкоджень апаратури. Завдяки ремонту і ТО вдається забезпечити необхідний рівень надійності обладнання протягом тривалого терміну експлуатації.

У процесі вивчення матеріалу даного розділу варто звернути увагу на різні види (плановий і позаплановий) ремонту апаратури. Позаплановий ремонт, як правило, проводять для усунення наслідків відмов раптового характеру або непроконтрольованих вчасно відмов поступового характеру. Плановий ремонт може регламентуватися наробітком апаратури і календарних термінів або може проводитися за результатами контролю стану функціонування обладнання.

Питання для самоперевірки

1. Які види ремонту використовуються при експлуатації засобів комп'ютерних систем і мереж?
2. Які складові визначають час відновлення працездатності?
3. Якими показниками характеризується процес відновлення працездатності обладнання?
4. Чому процес відновлення працездатності обладнання має випадковий характер?
5. Від яких чинників залежить відновлюваність обладнання КС та М?
6. Від чого залежить відновлюваність обладнання КС та М?
7. У чому особливості планового та непланового ремонту?
8. У якому випадку закон розподілу часу відновлення найточніше описується розподілом Ерланга другого порядку?
9. На які складові поділяється час ремонту?
10. Які закони розподілу випадкових величин використовують для аналітичного опису процесу відновлення обладнання?
11. Як впливає відновлюваність на надійність системи обладнання КС та М?

2.2.4. Контроль технічного стану КС та М

2.2.4.1. Загальні положення

Використання в різних галузях дедалі складніших комп'ютерних систем, зниження їх надійності в процесі експлуатації, а також необхідність забезпечення високої готовності до застосування вимагають постійного контролю їхнього технічного стану. Процес контролю технічного стану — це значна частка в загальному обсязі робіт з ТО і ремонту КС та М. Контроль здійснюють різними методами — вручну за допомогою контрольно-вимірювальних приладів або спе-

ціалізованих напівавтоматичних і автоматичних пристроїв. При цьому прагнуть забезпечити процес контролю таким чином, щоб контрольна апаратура не впливала на надійність контрольованих об'єктів.

Різні види контролю технічних засобів кваліфікують за рядом ознак. Так, за видом розв'язуваного завдання розрізняють контроль:

— функціонування — виконання об'єктом заданих функцій без їх якісної оцінки;

— працездатності — допусковий або кількісний контроль обраних визначальних параметрів;

— діагностичний — виконується задля локалізації місця ушкодження або відмови;

— прогнозний — для прогнозування стану об'єкта або його елементів протягом часу експлуатації;

— профілактичний — для виявлення і заміни елементів, параметри яких близькі до гранично припустимого.

За видом оцінки результату контролю:

■ допусковий контроль, здійснюваний для оцінки результату щодо встановленого рівня допуску на параметр;

■ кількісний контроль, що передбачає реєстрацію ступеня відхилення контрольованого параметра від номінального значення.

За ступенем зовнішніх впливів розрізняють контроль:

- пасивний — без зовнішнього впливу на об'єкт контролю;
- активний — із застосуванням зовнішніх стимуляторів.

За часом проведення контроль може бути:

✓ безперервний — у процесі роботи об'єкта;

✓ циклічний — при якому стан контрольованих параметрів аналізується в процесі роботи об'єкта через визначені інтервали часу;

✓ періодичний — здійснюваний для кожного з параметрів об'єкта через визначений період часу протягом заданого терміну експлуатації об'єкта.

За видом реалізації розрізняють контроль:

— ручний;

— автоматизований (за часткової участі людини);

— автоматичний.

За організацією розрізняють контроль:

◊ програмний — при використанні спеціальної програми для розв'язання контрольних завдань за допомогою тестів;

◊ програмно-логічний — заснований на використанні надлишкової вимірювальної інформації і проміжних результатів її обробки за спеціальною програмою;

◊ схемний — за допомогою спеціально вбудованого в об'єкт контролю обладнання;

◊ дистанційний;

◊ централізований — здійснюваний із загального пульта керування і контролю для сукупності розосереджених об'єктів.

Проведенню контролю повинен передувати самоконтроль апаратури контролю.

Одним із видів контролю є технічне діагностування, що, своєю чергою, може бути тестовим і функціональним. При тестовому діагностуванні на об'єкт від системи контролю надходять спеціальні тестові програми. У моменти тестового діагностування об'єкт звичайно не використовується за своїм призначенням. Функціональне технічне діагностування здійснюється в процесі безпосереднього використання об'єкта за призначенням.

Визначена сукупність тестових програм і послідовність їх виконання, що забезпечує діагностування, являють собою тест діагностування. Тест, використовуваний для перевірки працездатності об'єкта, називається перевірочним. Тест, використовуваний для локалізації місця відмови або ушкодження, називається тестом пошуку відмови, або розрізнювальним тестом. Алгоритм технічного діагностування задає сукупність елементарних перевірок, послідовність їх реалізації і правила обробки результатів контролю.

Зіставлення результатів контрольованих параметрів (КП) із визначеними областями їхніх значень, що відповідають конкретним технічним станам об'єкта, називаються діагностичними ознаками станів.

Залежно від природи контрольованих параметрів об'єкта контролю (ОК) розрізняють параметричні та фізичні методи діагностування. Параметричні методи ґрунтуються на контролі основних параметрів, що характеризують правильність функціонування об'єкта. Фізичні методи засновані на контролі характеристик тих явищ в об'єкті, що є наслідком його правильного або неправильного функціонування (нагрівання, електричні і магнітні поля, світлові випромінювання тощо).

При допусковому контролі розрізняють дво- й однобічні поля допусків. Для двобічного допуску установлюються верхня і нижня границі контрольованого параметра, за межі яких він не повинен виходити для збереження працездатного стану ОК. Для однобічного допуску регламентується тільки одна границя. У процесі експлуатації вихід параметра за границі допусків (границі працездатності) — неприпустимий. Тому для КП встановлюють більш вузьке поле випереджальних допусків, що створює запас працездатності.

При програмному контролі за допомогою тестів може бути прийнято одне з двох рішень: працездатний чи непрацездатний пристрій, тому допуски на параметри не встановлюють.

2.2.4.2. Показники діагностування

Для порівняльної оцінки різних методів і засобів контролю необхідно увести визначальні показники, що мають кількісне вираження. Через те, що системи діагностування мають методичні й інструментальні похибки, результати контролю можуть не відповідати реальному стану об'єкта. Тому одним з основних критеріїв оцінки ефективності діагностування є ймовірність правильності одержуваних при діагностуванні результатів або ймовірність правильного діагностування. Імовірність правильного діагностування залежить від кількості діагностичних параметрів, законів розподілу випадкових величин, що характеризують ці параметри, точнісних характеристик засобів вимірювання, можливості прихованих відмов і надійності засобів контролю.

Показники діагностування характеризуються також тривалістю, вартістю, трудомісткістю діагностування і глибиною контролю.

Середня оперативна тривалість діагностування визначається як математичне сподівання оперативної тривалості однократного діагностування.

Середня вартість діагностування визначається як математичне сподівання одноразового діагностування. Вона враховує амортизаційні витрати діагностування, витрати на експлуатацію системи діагностування і вартість зносу об'єкта діагностування.

Середня оперативна трудомісткість діагностування визначасться як математичне сподівання оперативної трудомісткості проведення одноразового діагностування.

Глибина контролю визначається складовою об'єкта, з точністю до якої визначається місце дефекту.

2.2.4.3. Шляхи удосконалювання методів контролю

Основними елементами системи контролю параметрів є:

— об'єкт контролю;

— контрольно-вимірювальна апаратура одержання, переробки й аналізу вимірювальної інформації;

— засоби передавання вимірювальної і керівної інформації;

— споживачі інформації.

Ці елементи утворюють локальну інформаційну систему контролю, використовувану для визначення поточного стану радіоелектронної апаратури в процесі експлуатації.

Неавтоматизовані системи контролю параметрів мають такі недоліки:

- низьку пропускну здатність контролю, малу швидкість;
- ручну реєстрацію результатів контролю;

- наявність суб'єктивних помилок операторів;
- невелику швидкість обробки результатів;
- великі працезатрати;
- надмірність різноманітної апаратури контролю;
- високу вартість контролю тощо.

Найвужчою ланкою, що обмежує можливості неавтоматизованого контролю, є оператор, особливо в разі виконання операцій вимірювання, обробки інформації й ухвалення рішення.

Перевагами створюваної автоматичної контрольно-вимірювальної апаратури є:

- велика пропускна здатність контролю;
- автоматичне документування результатів;
- виключення суб'єктивних помилок оператора;
- велика швидкість виконання всіх операцій контролю, обробки результатів і ухвалення рішення;
- скорочення кількості обслуговуючого персоналу;
- висока точність вимірів і ін.

Автоматизація контролю дає змогу скорочувати час його проведення в 20...30 і більше раз порівняно з неавтоматизованим процесом.

2.2.4.4. Характеристика об'єктів контролю

Об'єктом контролю є технічні пристрої, інформацію про технічний стан яких необхідно мати протягом усього процесу експлуатації. При експлуатації КС та М контролеві підлягають багато експлуатаційно-технічних характеристик, що вимагають різноманітних засобів вимірювання, обробки результатів і часто непростих операцій з ухвалення рішень. Оскільки апаратура використовується для виконання досить відповідальних завдань і не може бути вимкнена на тривалий час, необхідно вживати заходів для скорочення часу контролю і настроювання апаратів відповідно до вимог нормативно-технічної документації.

Під час контролю доводиться виконувати ряд операцій з вимірювання великої кількості параметрів різноманітного характеру, тому необхідно мати і різноманітну контрольно-вимірювальну апаратуру. Оскільки операції контролю досить трудомісткі, велике значення надається алгоритмові контролю, що визначає послідовність операцій, реалізовану для здійснення процесу контролю.

Тривалість і якість результатів контролю залежать від контролепридатності обладнання. Контролепридатністю називають властивість ОК, що характеризує його пристосованість до проведення контролю.

У технічних засобах КС та М параметри контролю є характеристиками електричного, радіотехнічного, електромеханічних і ряду інших процесів.

Розрізняють такі групи параметрів:

— вхідних і вихідних сигналів (амплітуда, афективне значення, тривалість імпульсів і їхніх фронтів, частота, потужність тощо);

— фізичних процесів, що протікають у самій апаратурі (напруга, струми, пульсації напруг, тривалість й амплітуда імпульсів, частота їх проходження, форма і т.д.);

— параметри без запасу енергії (коефіцієнт шуму, чутливість, вхідні і вихідні опори, параметри передатних і перехідних функцій і т.д.);

— визначальні вихідні (тактичні) характеристики окремих систем.

Розрізняють первинні, вторинні і проміжні контрольовані параметри. Первинні — це параметри елементів ОК, вони мають найнижчий ступінь узагальнення. Вторинні — параметри вихідних функцій об'єкта контролю. Вони мають найвищий ступінь узагальнення інформації про працездатність об'єкта і є визначальними. Проміжні — це параметри, які забезпечують зв'язок між вторинними і первинними параметрами.

Контрольовані параметри носять характер випадкових величин, оскільки залежать від впливу багатьох випадкових чинників (неточність виробництва, старіння, знос, зміна умов експлуатації, зміна напруги живлення, наявність перешкодової ситуації, зміна навантажень, відмови або ушкодження елементів систем і т.д.).

Поняття контролю параметрів містить вимірювання якої-небудь величини в кількісному вигляді і прийняття судження про працездатність об'єкта або стан даного параметра.

Результати контролю використовуються для подальшого впливу на об'єкт проведенням регулювань, заміною елементів, доробок.

До складу КС та М входять радіолокаційні і радіопеленгаційні системи, апаратура обробки інформації аналогового і цифрового характеру, обчислювальні комплекси, різна апаратура відображення інформації про повітряну обстановку і плани польотів, апаратура зв'язку, контролю і керування, документування різних процесів, електроживлення, кондиціонування тощо.

Працездатність такої різноманітної апаратури характеризується великою кількістю параметрів. У такій апаратурі протікають струми надвисоких, високих, проміжних і низьких частот, постійні струми. Вимірюються електромагнітні поля високого і низького рівнів напруги. Споживані і випромінювані потужності можуть вимірятися десятками кіловатів і навіть мегават, а потужність сигналів — менш ніж мільйонними частками мікровата. Тривалість вимірюваних сигналів і їхніх фронтів може змінюватися в дуже широких межах, починаючи з часток мікросекунди і вище.

Працездатність складних систем характеризується багатьма параметрами (первинними, проміжними і вторинними). Контролювати всі параметри складно і часто в цьому немає необхідності. Звичайно контролюють відносно невелику кількість визначальних і допоміжних параметрів. Установлення складу визначальних параметрів і їхніх допусків часто являє собою складне завдання дослідницького характеру.

При пошуку ушкодження необхідно контролювати більшу кількість параметрів, аніж при контролі працездатності. Для прогнозування працездатності необхідно мати ще більший обсяг інформації, тобто кількість контрольованих параметрів визначається завданнями контролю.

Для контролю стану експлуатаційно-технічних характеристик складних систем необхідно вибирати параметри більш високих ступенів узагальнення. Це можливо тільки за досить великої інформації, внесеної кожним первинним параметром в інформацію про параметр більш високого ступеня узагальнення. Тоді відхилення за межі встановлених допусків параметрів низьких ступенів узагальнення будуть впливати на фіксацію значень параметрів високих ступенів узагальнення.

2.2.4.5. Види систем контролю

У процесі експлуатації технічних засобів КС та М для контролю їх стану використовується різна вимірювальна апаратура загального застосування і спеціалізована.

Як вимірювальну апаратуру загального застосування використовують: тестер, стрілочні і цифрові прилади (амперметри, вольтметри, омметри, вимірники нелінійних викривлень), осцилографи, лічильники імпульсів, надвисокочастотні, високочастотні і низькочастотні генератори, частотоміри, аналізатори спектра, вимірники поля, вимірники часу й інші. Це — автономна контрольно-вимірювальна апаратура, керування її роботою і проведення вимірювань здійснюється, як правило, вручну.

Промисловість випускає і спеціалізовану апаратуру для вимірювання тих або інших параметрів і характеристик обладнання комп'ютерних систем.

Класифікація апаратури контролю здійснюється за такими ознаками:

— способом керування процесом контролю;

— видом зв'язку апаратури контролю з контрольованим об'єктом;

— принципом побудови апаратури контролю;
— видом обробки вимірюваної інформації;
— цільовим призначенням апаратури контролю;
— ступенем універсальності апаратури контролю;
— видом подання результатів контролю;
— видом програми керування процесом контролю.

За способом керування процесом контролю розрізняють апаратуру ручного, автоматизованого й автоматичного контролю. Апаратура автоматизованого контролю припускає часткову участь людини в процесі контролю, при автоматичному контролі участь людини виключається, така апаратура є програмно-керованою.

За видом зв'язку контрольно-вимірювальної апаратури з об'єктом контролю розрізняють автономну та вбудовану апаратуру. Автономна апаратура входить до складу обладнання об'єкта контролю.

За принципом побудови розрізняють дискретну, аналогову і змішану апаратуру контролю.

У дискретній апаратурі контролю процес вимірювання здійснюється в дискретному коді, усі вимірювані сигнали перетворюються в необхідний код, найчастіше у двійковий. У такій апаратурі синхронізуючі, керівні і контрольні сигнали також подаються у двійковому коді, зручному для роботи цифрових обчислювальних ЕОМ. Ця апаратура має високу швидкодію, дуже високу точність обробки вимірюваної й надаваної інформації, легку реалізацію автоматичного програмно-керованого контролю.

В аналоговій контрольно-вимірювальній апаратурі робота усіх функціональних вузлів здійснюється з безперервними сигналами. У такій апаратурі використовуються принципи роботи аналогової моделюючої апаратури.

У змішаній контрольно-вимірювальній апаратурі частина функціональних вузлів працює в дискретному коді, а інша частина — з безперервними електричними сигналами.

За видом обробки вимірюваної інформації розрізняють апаратуру з дискретною й аналоговою обробкою.

Дискретна обробка являє собою перетворення усієї вимірюваної інформації в дискретний код. У цьому коді відбувається виконання всіх логічних і обчислювальних операцій, необхідних для формування якісної або кількісної оцінки контрольованих параметрів і реєстрації результату контролю. Як вихідні реєструвальні пристрої такої апаратури використовують перфорувальні друкуючі та пристрої запису на магнітну стрічку.

Аналогова обробка вимірюваної інформації припускає перетворення всієї інформації в аналоговий вигляд, виконання всіх логічних операцій

з формування якісної і кількісної оцінки, реєстрацію результатів контролю також в аналоговому вигляді. Як реєструвальні пристрої використовуються стрілочні прилади, самописні реєструвальні пристрої тощо.

Змішаний вид обробки являє собою поєднання дискретної й аналогової обробок вимірюваної та подаваної інформації.

Відповідно до цільового призначення апаратура контролю може застосовуватися для розв'язання завдань контролю стану, прогнозування стану, пошуку відмов, автоматичної корекції заданих параметрів, визначення роботоздатності й ін.

За видом програми керування процесом контролю стану об'єкта розрізняють контрольно-вимірювальну апаратуру з зовнішньою й внутрішньою програмою.

Апаратура контролю з зовнішньою програмою характеризується тим, що програма контролю існує окремо від апаратури і може вводитися безпосередньо перед початком контролю та вимірюватися залежно від завдань контролю. Як носії програми використовують перфострічку, перфокарту і магнітну стрічку.

Апаратура контролю з внутрішньою програмою характеризується тим, що програма закладена в довгостроковий запам'ятовувальний пристрій апаратури контролю. Як носії програми використовують перфострічку, перфокарту і магнітну стрічку.

За ступенем універсальності розрізняють спеціалізовані й універсальні системи контролю. Спеціалізовані системи контролю призначені для контролю стану об'єктів одного виду. Вони звичайно відносно прості. Універсальні системи контролю призначені для розв'язання ряду завдань контролю стану різних типів об'єктів контролю.

За видом подання результату контролю розрізняють апаратуру з якісним і кількісним поданням і реєстрацією результатів контролю.

2.2.4.6. Загальна характеристика засобів і методів діагностування цифрових пристроїв

Всі ушкодження, що виникають в ЕОМ і в інших цифрових пристроях, можуть бути кваліфіковані за тривалістю, зовнішнім проявом і причинами виникнення. На функціонування таких пристроїв впливають відмови, ушкодження і збої в роботі.

Звичайно ушкодження не порушує працездатності пристрою, проте, якщо ушкодження призвело до порушення працездатності, то воно називається відмовним. Відмови, що самоусуваються, називаються збоями. В ЕОМ найбільш ненадійними ланками є електромеханічні пристрої (пристрої введення-виведення, накопичувачі на маг-

нітних носіях). Виявлення відмов цих ланок не викликає великих утрат часу, однак час відновлення відносно великий, оскільки операції заміни зіпсованих вузлів досить працемісткі.

Сьогодні розрізняють програмне, апаратне і програмно-апаратне діагностування ЕОМ.

Апаратні пристрої діагностування не виконують операцій, властивих ЕОМ. Вони включаються в структуру ЕОМ додатково і функціонують незалежно від розв'язуваних ЕОМ задач. Оскільки швидкодія розв'язуваних в ЕОМ задач велика, то для обмеження поширення помилок, що виникають у результаті збоїв, необхідно безупинно і досить оперативно відстежувати їх виникнення. Для цього використовуються швидкодіючі пристрої, котрі виконують операції виявлення помилок зі швидкістю, що відповідає швидкості власне машинних операцій. Використовувані для цього апаратні пристрої здійснюють перевірку працездатності ЕОМ без зниження якості її функціонування.

Звичайно ж, апаратне діагностування ускладнює структуру ЕОМ, трохи знижує її надійність, збільшує вартість.

При програмному діагностуванні установлення факту працездатності ЕОМ і пошук ушкодження здійснюються за допомогою спеціальних програм. Програмне діагностування буває тестове і програмно-логічне.

За тестового діагностування використовуються спеціальні програми, складені у вигляді тестів, що дають змогу перевіряти елементи ЕОМ у визначеному їх сполученні. Оскільки ЕОМ виконує рішення стандартної задачі з відомим результатом — еталоном, розбіжність результатів обчислення з цим еталоном свідчить про наявність визначеного ушкодження, закодованого в тесті.

Якщо використовується комбінація програмного й апаратного діагностування, то таке діагностування називається програмно-апаратним. Його застосування дає змогу використовувати позитивні сторони апаратного і програмного способів діагностування, завдяки чому скорочується час пошуку й усунення ушкодження.

Вибір методу діагностування здійснюється за допомогою інтегральних критеріїв ефективності, що відбивають найбільш важливі показники. До таких показників належать:

— імовірність того, що час виявлення ушкодження буде менше заданого або дорівнює йому;

— імовірність того, що ушкодження буде виявлено правильно;

— тривалість однократного діагностування τ_g, що включає власне час діагностування і час виконання допоміжних операцій діагностування;

— глибина пошуку ушкодження, що характеризує точність визначення місця ушкодження;

— обсяг діагностичного ядра, що характеризує частину апаратури ЕОМ, справну до початку діагностування.

Нині основним методом діагностування ЕОМ є тестове, котре здійснюється за спеціальними програмами, що визначає послідовність і характер операцій. Тестове діагностування дає можливість робити оцінку працездатності і пошук ушкодження на всіх етапах експлуатації ЕОМ: у період налагодження й випробувань, у процесі використання ЕОМ за призначенням, під час проведення ТО.

За характером одержуваної в процесі діагностування інформації тести поділяються на дві групи:

- тести, за якими перевіряється відсутність або наявність ушкодження в ЕОМ (оцінка працездатності);
- тести, використовувані для локалізації ушкодження і визначення його характеру.

Недолікам тестового діагностування є можливість його використання тільки під час перерв у виконанні основних робочих функцій ЕОМ.

Розглянемо далі деякі особливості методів тестового діагностування.

Двоетапне діагностування — це метод тестового діагностування, застосовуваний для діагностування логічних схем з пам'яттю. Алгоритм діагностування містить результати тестового впливу й адреси всіх елементарних перевірок. Алгоритм діагностування має стандартний формат і називається тестом локалізації несправності (ТЛН). Уведення тестів, фіксації відповідей, аналіз і видача результатів реалізації алгоритму діагностування здійснюються за допомогою стандартних діагностичних операцій.

До складу діагностичного тесту входить настановна і керівна інформація, адреса комірки пам'яті для запису результату елементарної перевірки, еталонне значення реакції на тест, адреси ТЛН, яким передається керування при збігу або розбіжності результатів тесту з еталонними значеннями.

Послідовне сканування — це метод тестового діагностування, при якому, наприклад, регістри і тригери утворюють один регістр зсуву. При цьому можлива установка регістра в довільний стан і опитування за допомогою операції «зсуву».

Метод послідовного сканування застосовується при діагностуванні ЕОМ, побудованих на великих інтегральних схемах (ВІС). При побудові ЕОМ на ВІС виникають труднощі діагностування в

зв'язку з обмеженим доступом до схем усередині кожної ВІС. Метод послідовного діагностування вирішує цю проблему за невеликої кількості додаткових входів і виходів.

Мікродіагностування — це метод тестового діагностування апаратури для виконання мікрооперацій. У мікропрограму діагностування закладаються тести для перевірки мікрооперацій. Мікропрограма перевірки чергової мікрооперації використовує усі перевірені мікрооперації і тракти передачі інформації. При цьому методі за допомогою набору мікрооперацій, передбачених в ЕОМ, за наявними інформаційними трактами тестові програми надходять на вхід апаратури, що перевіряється. З виходу цієї апаратури сигнали надходять на спеціальні схеми, де відповіді порівнюються з еталонними або програмно, або ж за допомогою діагностичних операцій опитування і порівняння.

Мікродіагностика може бути вбудованою, коли діагностичні мікропрограми розміщаються в постійній мікропрограмній пам'яті, і завантажуваною, коли діагностичні мікропрограми розміщаються на зовнішньому носії даних. У сучасних ЕОМ мікродіагностикою охоплюються практично всі тригери і регістри.

Автоматичне тестове діагностування мікропроцесорів і мікро-ЕОМ утруднюється через велику складність і високу вартість апаратури тестового (і вбудованого) діагностування. Однак діагностування цієї апаратури необхідне, тому постає питання про можливості визначення ушкоджень ручним способом. При цьому на вхід пристрою подається послідовність вхідних сигналів, а потім вихідна послідовність сигналів порівнюється з еталонною, зазначеною в документації. Безліч різноманітних вузлів для діагностування, використовуваних у мікропроцесорах і мікро-ЕОМ, вимагають великої кількості різноманітних тестів.

Для підвищення ефективності діагностування мікропроцесорної техніки використовується **метод сигнатурного аналізу**, що полягає у застосуванні циклічних надлишкових кодів для стиснення довгих двійкових кодів, що характеризують реакцію апаратури на тести, у короткий, звичайно 4-, 5-розрядний шістнадцятковий код. Цей код легко відображається на індикаторах і порівнюється з контрольним кодом, зазначеним у технічній документації для кожної точки, що перевіряється. Контрольний код називається сигнатурою.

Циклічні коди засновані на зображені переданих даних у вигляді полінома і використовуються для послідовного обміну даними між ЕОМ і зовнішніми запам'ятовувальними пристроями, а також при передаванні даних по каналу зв'язку.

Контроль правильності передавання даних за допомогою циклічних (поліноміальних) кодів заснований на такій закономірності. Якщо інформаційний поліном $G(x)$ помножити на деякий так званий породжувальний поліном $P(x)$, а потім сформований кодовий поліном передати приймачеві інформації і виконати в ньому зворотну дію (ділення полінома прийнятого повідомлення на породжувальний поліном), то ненульовий залишок означатиме, що сталася помилка. Нульовий залишок означає, що помилки немає або вона не виявлена.

Уведемо такі позначення:

$G(x)$ — інформаційний поліном, що відповідає переданій інформації довжиною m біт. Він має ступінь менше m–1; $P(x)$ — породжувальний поліном ступеня k, що визначає кількість контрольних біт, а також виявляє і коректує здатність циклічного коду; $F(x)$ — кодовий поліном, який відповідає переданому циклічному кодові. Це поліном ступеня $(m + k)$, він поділяється без залишку на породжувальний поліном ступеня k.

Звичайно використовується формування кодового полінома, при якому старші коефіцієнти утворять інформаційні знаки, а молодші — контрольні. Інформаційний поліном $G(x)$ ступеня $(m - 1)$, який необхідно закодувати, множать на x^k, що відповідає зсуву на k розрядів ліворуч. Отриманий після цього поліном $x^k G(x)$ ділиться на поліном $P(x)$ для визначення залишку $R(x)$:

$$\frac{x^k G(x)}{P(x)} = Q(x) \oplus \frac{R(x)}{P(x)},$$

де $Q(x)$ — частка від операції ділення; $\oplus$ — знак, який позначає операцію додавання за модулем 2.

Звідси випливають вирази:

$$\begin{aligned} x^k G(x) &= Q(x)P(x) \oplus R(x); \\ F(x) &= Q(x)P(x) = x^k G(x) \oplus R(x). \end{aligned} \qquad (2.23)$$

При стисканні довгих двійкових кодів використовується сигнатурний аналізатор, побудований на основі зсувного регістра із внутрішніми зворотними зв'язками, що замикаються через суматор по модулю 2. На вхід суматора надходить послідовність імпульсів, що знімається в контрольній точці схеми. Для індикації показань у шістнадцятирозрядному коді зсувний регістр має індикатор.

Стиснення даних у сигнатурному аналізаторі відбувається так. На зсувний регістр з виходу схеми, що перевіряється, надходить

двійкова послідовність імпульсів у вигляді інформаційного полінома $G(x)$. Вона зображується у вигляді полінома $x^k G(x)$, де k — кількість розрядів зсувного регістра. Далі ця послідовність поділяється на породжувальний поліном $P(x)$ ступеня k. Цей розподіл реалізується на регістрі зсуву зі зворотними зв'язками. Залишок від розподілу $R(x)$ зберігається в регістрі. Аналітично ця операції виражені залежністю (2.23).

При проходженні послідовності x у процесі розподілу залишок $R(x)$ змінюється до закінчення всієї послідовності x. Кінцевий вираз $R(x)$ є сигнатурою.

Сигнатурний аналіз відрізняється високою швидкодією, зумовленою швидкодією зсувного регістра і суматора по модулю 2. Йому властива висока вірогідність, тому що контрольні суми вихідних двійкових розрізнюваних сигналів мають різні сигнатури.

Діагностування внутрішніх ланцюгів передавання інформації в обчислювальних комплексах засновано на використанні інформаційної надмірності, тобто кодів з виявленням і корекцією помилок. Ці коди допускають операції перевірок на парність і непарність.

Найпоширенішим методом діагностування суматорів є контроль парності. Його сутність полягає в такому. При додаванні чисел a_i і b_i, $1 \le i \le n$, розряди суми у воряться відповідно до виразів:

$$S_1 = a_1 \oplus b_1;$$
$$S_2 = a_2 \oplus b_2 \oplus c_2;$$
$$\ldots\ldots\ldots\ldots\ldots\ldots$$
$$S_n = a_n \oplus b_n \oplus c_{n-1}.$$

Тут C_n — число, що відповідає сигналам переносу в старший розряд. Склавши всі n рядків по модулю 2, одержимо:

$$P_S = p_a \oplus p_b \oplus p_c,$$

де P_S — парність суми; p_a, p_b — парність доданків; p_c — парність переносу.

Таким чином, контроль парності суми може бути здійснений через порівняння розрахованої парності за формулою (2.23) з дійсною парністю суми.

Цей спосіб дає змогу виявляти одиничну помилку і будь-яке непарне число помилок.

Методи апаратного і перевірочного діагностування часто застосовуються в поєднанні, тобто у вигляді програмно-апаратних методів діагностування.

2.2.4.7. Автоматичний збір та обробка інформації про надійність КС та М

Для аналізу ефективності роботи обчислювальних засобів й інших пристроїв в КС та М передбачені засоби автоматичного накопичення й обробки інформації про надійність їхньої роботи. Накопичуються, наприклад, статистичні дані про помилки задля виявлення найбільш імовірних джерел помилок і локалізації причини збоїв. Це одна з найбільш складних проблем експлуатації комп'ютерних систем унаслідок невідтворюваності ситуації і непередбачуваності подій.

Автоматичне накопичення інформації про помилки відбувається шляхом реєстрації стану комп'ютерних систем в момент помилки. В операційних системах використовуються спеціальні засоби обробки помилок за типами і накопичення інформації про них у спеціальних системних журналах помилок (це звичайно область пам'яті резидентного диска).

Необхідно прагнути охопити всі можливі помилки в системі, зареєструвавши їх і зберігши задля наступного аналізу і скорочення витрат часу операційної системи на ці операції.

Використовуються такі програмні засоби обробки інформації про помилки:

— оброблювачі:
— машинних помилок;
— канальних помилок;
— збоїв периферійних пристроїв;
— відмов периферійних пристроїв;
— перевантажень операційної системи;
— утрат переривань уведення-виведення;
— реконфігурації;
— програмних помилок.

Оброблювач машинних помилок обробляє помилки, що виявляються схемами контролю процесора. Він реєструє стан ЕОМ у момент помилки процесора в системному журналі і забезпечує відновлення системи.

Оброблювач канальних помилок реєструє інформацію про помилку в системному журналі і готує повторення команди уведення-виведення, що відмовила, з допомогою програм відновлення, що є для кожного периферійного пристрою.

Для реєстрації стану ЕОМ у момент помилки він заморожується, записується в оперативну або спеціальну пам'ять (апаратурна реєстрація), а потім — у системний журнал помилок.

Обсяг апаратурної реєстрації визначає точність аналізу і локалізації помилок. Апаратурна реєстрація залежно від структури може бути об'єднаною або роздільною. При сполученні апаратури процесора і каналів — реєстрація об'єднана, при незалежній структурі — роздільна.

Реєстрація можлива в оперативній або спеціальній буферній пам'яті. Реєстрація в оперативній пам'яті здійснюється за дозволу переривання від схем контролю. Однак, якщо в момент помилки переривання неприпустиме, то інформація про стан ЕОМ у момент помилки губиться. Особливістю реєстрації в спеціальну буферну пам'ять є те, що вона дає змогу зберігати інформацію про помилки навіть у разі заборони переривання від схеми контролю. Перепис зареєстрованої інформації станів ЕОМ у системний журнал помилок здійснюється оброблювачем машинних помилок за наявності дозволу на переривання від схем контролю.

Носіями даних про помилки при реєстрації інформації є стандартні магнітні диски, касетні магнітні стрічки або гнучкі магнітні диски пультових накопичувачів, магнітні стрічки «накопичення» і магнітні стрічки «історії».

При роботі операційної системи ЕОМ інформація про її стан у момент помилки записується в системному журналі помилок на стандартних магнітних дисках, однак при цьому потрібна працездатність більшої частини обладнання ЕОМ. Інакше виникає небезпека того, що система не може записувати інформацію про стан ЕОМ у системний журнал помилок. Тому для запобігання втратам інформації про помилки поряд із записом у системний журнал ведеться також реєстрація інформація про стан ЕОМ на магнітні носії пультових накопичувачів за допомогою апаратних засобів. Для цієї мети використовується тільки апаратура діагностичного ядра ЕОМ.

Накопичення й обробка інформації про стан ЕОМ у моменти помилок необхідні для розробки заходів щодо підвищення надійності ЕОМ. Записи про помилки накопичуються і при проведенні профілактичних випробувань ЕОМ.

Інформація про помилки звичайно записується в накопичувачах і зберігається невеликий проміжок часу. За тривалий період експлуатації ЕОМ інформація про помилки накопичується на стрічках історії. Для цього призначені спеціальні програмні засоби перепису на стрічку історії інформації про помилки із системного журналу, магнітних носіїв накопичувачів.

Для збирання статистичних даних про збої кожного з периферійних пристроїв ЕОМ служить реєстратор збоїв периферійних пристроїв. Він одержує команди керування з програм відновлення від збоїв периферійних пристроїв у складі операційної системи, кожен збій периферійного пристрою кількісно реєструється лічильником збоїв. При переповненні лічильника збоїв реєстратор збоїв записує інформацію про помилку периферійного пристрою в системний журнал помилок.

Інформація про відмови периферійного пристрою (про непоправну помилку) за допомогою реєстратора периферійних пристроїв записується в системний журнал помилок.

Реєстратор перезавантажень операційної системи забезпечує запис у системний журнал помилок інформації про факт і причину перезавантажень. Кількість перезавантажень операційної системи є одним із показників надійності системи. У ряді випадків перезавантаження операційної системи призводить також до її відмови. Причинами перезавантаження операційної системи можуть бути несправність живлення, помилки системних програм, помилки в роботі апаратури, носія даних, оператора, у програмі користувача й ін.

Обробка статистичних даних із причин перевантажень операційної системи дає змогу встановлювати найбільш імовірні причини і вживати заходів з їх усунення.

У системний журнал інформації записується інформація про наявність або відсутність переривань уведення-виведення за допомогою реєстратора втрат переривань уведення-виведення. При цьому записується тип периферійного пристрою, його адреса і час. У системний журнал також записується інформація про випадки динамічної реконфігурації пристроїв (за допомогою реєстратора) і про програмні помилки, деякі помилки оператора, невдалі спроби програмного відновлення в разі помилок в апаратурі (за допомогою реєстратора програмних помилок).

В ЕОМ передбачаються програмні засоби накопичення, редагування і друкування інформації про помилки. Вони містять незалежну системну програму, що працює в автономному режимі, а системна — під керуванням операційної системи.

Незалежна програма виконує перепис зареєстрованої інформації з магнітних носіїв пультових накопичувачів на стрічку накопичення, редагування і друкування реєстрації.

Системна програма виконує редагування і друкування, узагальнення помилок за типами і пристроями, перепис інформації про помилки на стрічку історії. Вона дає змогу одержувати:

— узагальнені дані про відмови і збої ЕОМ за заданий інтервал часу;

— узагальнені дані про відмови і збої периферійних пристроїв;

— характер зміни надійності ЕОМ протягом заданого періоду експлуатації з указівкою кількості відмов і збоїв за кожний день заданого інтервалу часу;

— узагальнення статистичної інформації з усіх накопичувачів інформації;

— роздруківку історії відмов і збоїв за обраний етап експлуатації;

— роздруківку за типами записів.

Для автоматизованої обробки інформації про надійність існують пакети прикладних програм. Система автоматизованої обробка інформації про надійність дає змогу створювати банк симптомів помилок, ушкоджень і відмов ЕОМ й іншої апаратури, рекомендацій з їх усуненню. З її допомогою можна вирішувати завдання прогнозування надійності, замовлень на запасні елементи тощо.

Методичні вказівки

Чим складніше система, тим більшого значення набувають питання автоматизації контролю її працездатності. Неавтоматизований контроль звичайно вимагає значних витрат робочого часу.

В КС та М широко використовується допусковий контроль вихідних параметрів апаратури, програмно-логічний і програмний контроль. Застосовується як тестове, так і функціональне діагностування.

В апаратурі КС та М дуже поширені цифрові інтегральні схеми. При розробці перспективної техніки характерний перехід до інтегральних схем високих рівнів інтеграції, що накладає визначені умови на побудову систем контролю.

При вивченні даного розділу доцільно звернути увагу на системи контролю працездатності таких КС, які входять до складу сучасних автоматизованих систем управління технологічними процесами, до яких можна віднести автоматизовані системи управління повітряним рухом. Слід чітко зрозуміти, які можливості для обслуговування за станом надає використовуваний в КС та М сигнатурний контроль, як впливає цей контроль на локалізацію ушкоджень.

В міру ускладнення КС та М, як правило, ускладнюється й апаратура автоматичного контролю і керування конфігурацією системи. Відмови апаратури контролю можуть впливати на працездатність контрольованого обладнання. Щоб цей вплив зменшити, апаратура контролю доповнюється пристроями самоконтролю. Варто знати, до чого можуть призвести відмови апаратури контролю в КС та М.

Питання для самоперевірки

1. Дайте класифікацію видів контролю технічних засобів.
2. Якими показниками характеризується якість діагностування стану технічних засобів КС та М?
3. У чому полягають основні недоліки неавтоматизованих систем контролю?
4. Які основні параметри технічних засобів підлягають контролеві в КС та М?
5. З яких міркувань вибираються допуски на контрольовані параметри?
6. Які види систем контролю ви знаєте?
7. У чому полягають основні особливості систем діагностування пристроїв на цифровій елементній базі?
6. Як діагностується функціонування ЕОМ?
9. Схарактеризуйте методи тестового діагностування ЕОМ.
10. Що являють собою контроль парності інформації і контроль по модулю 2?
11. Як здійснюється діагностування периферійних пристроїв ЕОМ?
12. У чому полягають особливості діагностування засобів передавання інформації в КС та М?
13. Як доцільно обробляти інформацію про надійність роботи технічних засобів КС та М?
14. Що дає для служби експлуатації наявність систем контролю працездатності апаратури КС та М?

2.3. Організація високопродуктивних обчислювальних структур у комп'ютерних системах критичного застосування

2.3.1. Вимоги до обчислювальних мереж для комп'ютерних систем критичного застосування

Головним завданням мережі є забезпечення можливості спільного використання ресурсів у реальному часі. Для того, щоб мережа успішно справлялася з цим, вона має відповідати вимогам продуктивності, надійності й ін. Конкретизуємо ці вимоги стосовно задачі забезпечення роботи системи критичного застосування, до якої можна

віднести автоматизовану систему управління повітряним рухом (АС УПР). У процесі функціонування АС УПР в разі великих перепадів інтенсивності повітряного руху можуть спостерігатися різкі перепади обсягів інформації, яка передається по комп'ютерних мережах.

Продуктивністю мережі визначаються обсяг переданих даних і час, необхідний на їх передавання. Для оцінки продуктивності мереж загального призначення використовуються загальноприйняті числові характеристики — час реакції мережі, середня пропускна здатність, максимально можлива пропускна здатність, затримка передавання. Стосовно систем критичного застосування пропускна здатність задається, безпосередньо виходячи з максимальної очікуваної швидкості протікання процесів у системі, що обслуговується. Надалі вона може змінюватися (як правило, убік збільшення) при зміні вимог до системи в цілому. Наприклад, для АС УПР важливим чинником є максимальна очікувана інтенсивність повітряного руху в зоні відповідальності. Однак при цьому час реакції і затримка передавання даних можуть мінятися в широких межах, що неприпустимо при обслуговуванні повітряного руху, особливо в разі виникнення позаштатних ситуацій. Тому пріоритетними вимогами до обчислювальної мережі є гранично припустимі значення саме часу реакції і затримки передавання даних.

Надійність означає ймовірність того, що мережа виконує свої функції. Надійність технічних пристроїв звичайно характеризується *часом наробітку на відмову* і *коефіцієнтом готовності* (відсоток часу, протягом якого система може бути використана). Надійність інформаційно-комунікаційних мереж також характеризується ймовірністю доставки повідомлення адресатові. Надійність і ремонтопридатність мереж систем критичного застосування мають бути такими, щоб у разі відмов елементів зниження характеристик системи не виходило за деякі заздалегідь визначені межі. Час відновлення устаткування або програмного забезпечення не повинен перевищувати десятків секунд, максимум одну-дві хвилини.

Безпека означає захист від несанкціонованого доступу до даних і забезпечення надійності і стійкість до навмисних руйнівних впливів.

Розширюваність — це можливість порівняно легкого додавання нових елементів мережі. Для систем критичного застосування висувається додаткова вимога — можливість модифікації мережі в процесі її функціонування, без зниження експлуатаційних характеристик.

Масштабованість — це можливість нарощування розмірів мережі, у тому числі через приєднання додаткових сегментів.

Прозорість означає можливість використання ресурсів мережі тим самим способом незалежно від їх фактичного розміщення — на

локальному комп'ютері або в мережі. При цьому користувач ніби «не зауважує» мережі, працюючи безпосередньо з ресурсами.

Підтримка різних видів трафіка — можливість сполучення функцій різних мереж, наприклад телефонної, зв'язкової та комп'ютерної.

Керованість — можливість централізованого виявлення й усунення збоїв, несправностей, розподілу ресурсів і повноважень між користувачами. Зупинимося на цій задачі докладніше, оскільки для великої просторово розподіленої корпоративної мережі критичного застосування ефективність керування має найважливіше значення.

Для керування корпоративними комп'ютерними мережами, що включають велику кількість активного устаткування, необхідні складні системи керування, що здійснюють моніторинг, контроль і керування кожним елементом комп'ютерної мережі.

Існуючі системи керування, незважаючи на їхню функціональну надмірність, не мають у своєму складі розвинутих інтелектуальних засобів, які б давали змогу якісно прогнозувати поводження комп'ютерної мережі. Більшість засобів керування насправді мережею не керують, а всього лише пасивно здійснюють її моніторинг. Вони стежать за мережею, але не виконують активних дій, при цьому фіксуючи тільки факт збою. Ідеальним рішенням була би розробка системи аналізу, прогнозування і локалізації можливих збоїв у роботі як комп'ютерної мережі в цілому, так і окремих її елементів. Така властивість системи допоможе заздалегідь виявити можливі «вузькі» місця і вжити заходів щодо завчасної їх ліквідації. Однак така система керування практично нереалізовна для роботи в умовах критичного застосування. Тому реальним підходом до вирішення даної задачі видається поточна адаптація деяких підсистем системи в цілому до змінних умов застосування, перерозподіл ресурсів мережі для розв'язання конкретних пріоритетних задач (наприклад, у разі виникнення екстремальних ситуацій різного характеру). Такий підхід цілком логічний і природний, якщо врахувати, що будь-яка велика корпоративна мережа складається з окремих сегментів, котрі порівняно слабко впливають один на одного.

Проаналізуємо характеристики очікуваного навантаження на інформаційно-обчислювальну мережу АС УПР, що має працювати в реальному часі й в умовах можливого виникнення екстремальних ситуацій, тобто в умовах критичного застосування. На основі проведеного аналізу можна розробити метод і побудувати алгоритм адаптації фрагментів мережі до змін обсягів перероблюваної інформації та режимів роботи системи в цілому.

2.3.2. Аналіз навантаження на обчислювальні мережі автоматизованих систем керування повітряним рухом

Актуальність і перспективність об'єднання мереж різного призначення, надання через ту саму мережу послуг різного характеру, гармонізації комп'ютерних і телекомунікаційних технологій уже не викликають сумнівів. Передбачається, що будуть інтегруватися не тільки мережі різного цільового призначення (комп'ютерні, телекомунікаційні, документального електрозв'язку), а й мережі з різними принципами побудови — з фіксованим (провідний і безпровідний) і рухомим (безпровідний зв'язок) методами комутації.

Вважається також, що внаслідок постійної зміни співвідношення між обсягами телефонного трафіка і трафіка передавання іншої інформації (на користь другого) роль інфраструктури передачі даних зростатиме, а організація послуг (зокрема, додаткових видів обслуговування) дедалі менше буде зв'язана власне з транспортом інформації. Як відомо, такий підхід є основним при побудові інтелектуальних мереж (ІМ). Тому можна стверджувати, що перелічені тенденції розвитку — взаємозалежні складові загального процесу модернізації існуючих мереж і впровадження мереж нових поколінь в усіх сферах діяльності людини, зокрема, і в мережах авіаційного електрозв'язку, системах організації повітряного руху.

Основним принципом побудови як наявних, так і перспективних (інтегрованих) мереж об'єктивно є модульний принцип. Це зумовлено безліччю історичних, організаційних, технічних, виробничих і інших чинників. Характеристики кожного модуля, параметри систем керування і сигналізації узгоджуються між собою, і модулі поєднуються в корпоративні, регіональні і загальнонаціональні мережі.

Насамперед дамо визначення мереж нових поколінь. Одна з основних цілей їх реалізації — це передавання й обробка різнорідного трафіка (мова, дані, відео) з якістю, що забезпечується в мережах із комутацією каналів, у яких надаються послуги так званого операторського класу (з коефіцієнтом готовності «п'ять дев'яток», тобто не більш одного відмовлення в обслуговуванні протягом року). У цьому сенсі можна говорити про пресловуту «конвергенцію мереж», під якою мається на увазі надання можливостей обміну інформацією між різними мережами:

— телефонними мережами загального користування (ТФЗК);
— інтелектуальними мережами;
— мережами транкінгового і мобільного зв'язку;

— мережами радіозв'язку, у тому числі космічного зв'язку;
— *IP-мережами*.

Відповідно до моделі інформаційно-комунікаційної системи, запропонованої Міжнародним союзом електрозв'язку (МСЕ), характерною її особливістю є поділ рівнів термінального устаткування клієнта, мереж доступу і транспортних мереж, засобів сигналізації і керування, засобів створення послуг (рис. 2.11).

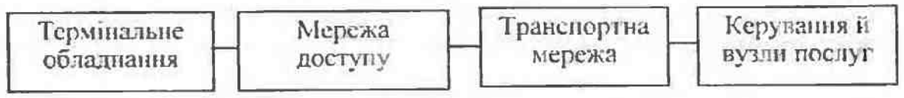


Рис. 2.11. Модель інформаційно-комунікаційної системи

В організаційному плані в такій системі повинна забезпечуватися підтримка всього спектра послуг, як на рівні транспортних мереж, так і на рівні мереж доступу для терміналів, що знаходяться у розпорядженні клієнта в даний час, — звичайних, мобільних або *IP-телефонів*, внутрішніх мереж гучномовного зв'язку, персональних комп'ютерів. Окрім того, за заявками клієнтів необхідно швидко і без зупинки роботи системи замовника (у нашому випадку — системи управління повітряним рухом (УПР)) організовувати і / або модифікувати індивідуальні набори послуг, у тому числі і додаткові види обслуговування. Іншими словами, повинна забезпечуватися так звана *мультисервісність*.

Деякі фахівці вважають, що конвергентні мережі — проміжний етап на шляху до мультисервісних мереж. Іноді і ті, й інші мережі називають «мережами наступного покоління» (*NGN — Next Generation Networks*). Однак у будь-якому разі мережі нових поколінь характеризуються такими принциповими особливостями:

- багатошарова інфраструктура з кількістю незалежних шарів від чотирьох до шести (за оцінками різних фахівців), причому кожний з них може створюватися незалежно від інших за аналогією з сталонною моделлю відкритих систем;
- наявність відкритих інтерфейсів і стандартних протоколів обміну між апаратурою доступу, комутації, керування і сигналізації;
- підтримка старих і створення нових послуг з універсальним доступом з будь-якої підмережі того або іншого виду — конвергенція послуг зв'язку;
- незалежність технологій створення апаратури і розробки програмного забезпечення (ПЗ) від технологій передавання й обробки даних;
- вирішення проблем сигналізації та керування на якісно новому рівні;

• підтримка технології комутації пакетів зі збереженням протягом деякого, можливо, досить тривалого періоду, технології комутації каналів.

Розглянемо параметри і статистичні характеристики трафіка конвергентних мереж. Як уже зазначалось, у загальному різнорідному трафікові (мова, дані, відео, інші типи) частка мовного трафіка знижується, причому темп цих змін наростає з року в рік. Наприклад, у США і Західній Європі щорічний темп росту трафіка даних становить до 30 % на рік, у той час як ріст телефонного трафіка — близько 3 %.

Перерозподіл видів навантаження на телекомунікаційні мережі має такі наслідки:

1. Поява другої моди в розподілі тривалості телефонного з'єднання: якщо для телефонних переговорів це 3...5 хв, то для користувача Інтернету — 20 хв (за іншими оцінками — 40 хв). У корпоративних мережах тривалість користування Інтернетом може бути ще більшою.

2. Зміна статистичних характеристик трафіка. Для телефонних мереж широко застосовуються марківські моделі, потоки Пуассона й Ерланга. У той же час результати численних експериментальних досліджень трафіка даних свідчать про те, що він має досить специфічні властивості і не може бути задовільно описаний у рамках класичної теорії масового обслуговування.

Для моделювання трафіка комп'ютерних мереж використовують різні сучасні методи.

Трафік даних, що циркулює в цифрових мережах, і, зокрема, у мережах з комутацією пакетів, має самоподібні, або фрактальні, властивості. «Самоподібність» являє собою властивість процесу зберігати своє поводження і зовнішні ознаки при розгляді в різному масштабі. Для часових послідовностей масштабованою величиною є час. Виходячи з визначення самоподібності, можна стверджувати, що часові і спектральні характеристики випадкового процесу (у нашому випадку — трафіка) при зміні масштабу усереднення будуть описуватися тими самими рівняннями, функціями, але з відповідними масштабними коефіцієнтами. Іншими словами, самоподібність якого-небудь процесу (явища) можна трактувати як інваріантість до змін масштабу або розміру.

Математичною основою самоподібних процесів є замкнуті множини, зокрема так звана *канторова множина*. Розглянемо структуру канторової множини. Нехай T_0 — відрізок одиничної тривалості: $T_0[0;1]$. Множина T_0 має свої кінцеві точки (є замкнутою). Викинемо з неї інтервал $(1/3, 2/3)$, тобто всі точки, що належать до середньої частини відрізка T_0, за винятком кінцевих. Множина, що залишилася, $T_1[[0,1/3],[2/3,1]]$, також має свої кінцеві точки, отже, також є замкнутою. Потім викинемо з T_1 інтервали $(1/9, 2/9)$ і $(7/9, 8/9)$, а множину, що залишилася й також буде замкнутою, позначимо T_2. В кожному з отриманих чотирьох відрізків завдовжки $(1/3)^2 = 1/9$ викинемо середній інтервал завдовжки $(1/3)^3 = 1/27$ і т. д. Продовжуючи цей процес, одержимо спадну послідовність замкнутих множин T_n. При $n \to \infty$ з відрізка $[0,1]$ видаляється злічена кількість інтервалів змінної (кратної) довжини. Дістаємо замкнуту множину $T = \bigcap_{n=0}^{\infty} T_n$. На рис. 2.12 зображено кілька етапів «проріджування» відрізка T_0.

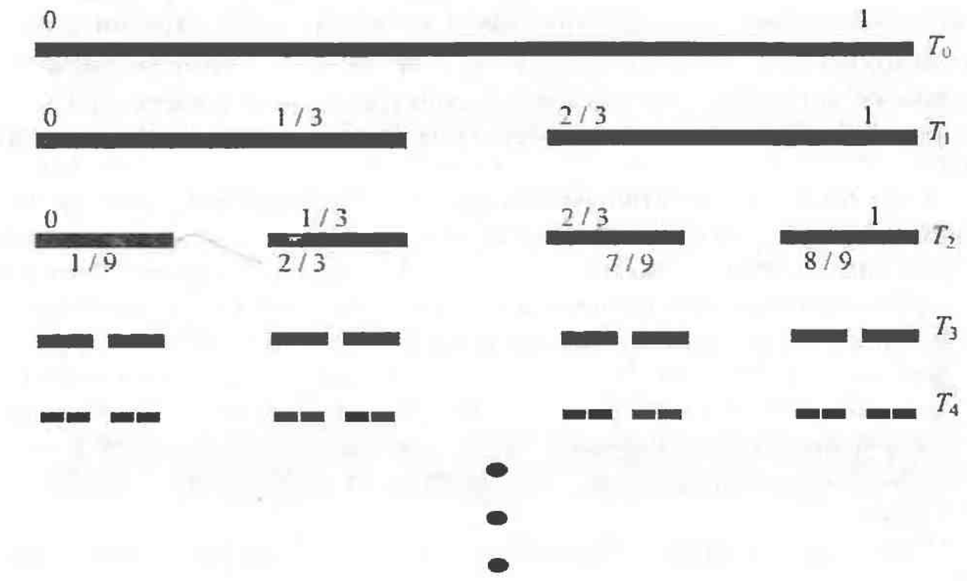


Рис. 2.12. Варіант канторової множини

Реальні випадкові процеси, звичайно, зберігають властивість самоподібності тільки до певної межі. Ця межа або міра статистичної стійкості процесу при багаторазовому масштабуванні визначається так званим параметром Херста або параметром самоподібності. Випадковий процес $x(t)$ є статистично самоподібним із параметром Херста H ($0{,}5 \le H \le 1$), якщо для будь-якого речовинного значення $a > 0$ процес $x(at)/a^H$ має ті самі статистичні характеристики, що і процес $x(t)$:

математичне сподівання $M[x(t)] = \frac{M[x(at)]}{a^H}$;

дисперсія $D[x(t)] = \frac{D[x(at)]}{a^{2H}}$;

кореляційна функція $R(t,\tau) = \frac{R(at,a\tau)}{a^{2H}}$.

Чим більше H, тим довше зберігається властивість самоподібності при багаторазовому масштабуванні. При $H = 0{,}5$ ця властивість практично відсутня.

Кореляційні функції самоподібних процесів із великим параметром Херста загасають повільніше, ніж у звичайних випадкових процесах, причому мають, як правило, коливальний характер. Установлено, що спадання постійної складової кореляційної функції відбувається за законом $c_1 t^{-c_2 a}$, де c_1, c_2 — константи, a — параметр масштабу. Відповідно і спектральна щільність процесу теоретично прямує до нескінченності при частоті, що прямує до нуля.

Для опису щільностей імовірностей самоподібних потоків використовують розподіли з «важкими хвостами»: логарифмічно-нормальне, гамма-розподіл, розподіли Вейбулла, Парето. Останнє використовується для опису самоподібного трафіка найчастіше. З усіх повільно загасаючих розподілів воно описується найпростішими математичними формулами. (Звичайно, простота не може бути підставою для використання тієї або іншої моделі процесу, тому надалі необхідно буде перевіряти належність вибірок самоподібного процесу до генеральної сукупності з тим або іншим імовірнісним розподілом.)

Вираз для щільності ймовірності розподілу Парето має такий вигляд:

$$f(x) = \frac{\alpha}{k}\left(\frac{k}{x}\right)^{\alpha+1},$$

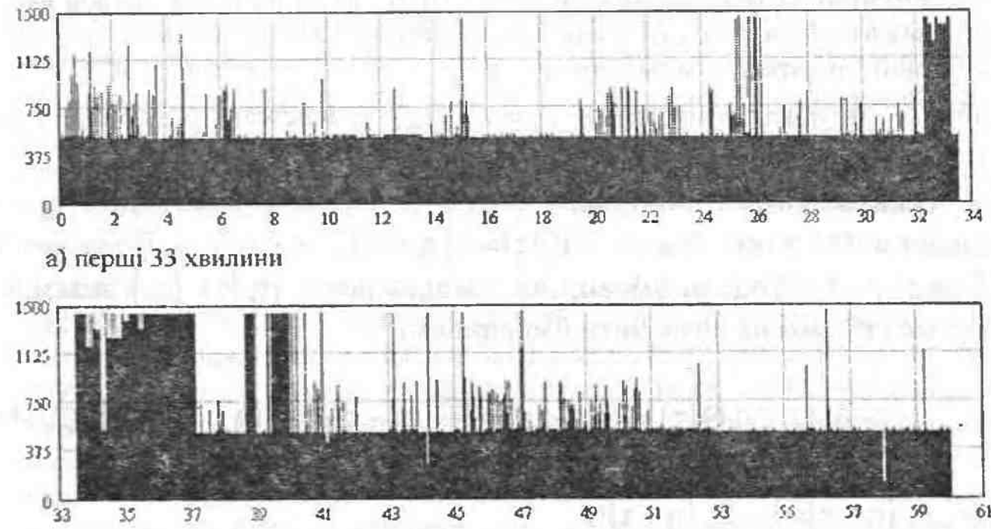
де k і α (k, $\alpha < 0$) — параметри розподілу.

Відповідно функція ймовірності $F(x) = 1 - \left(\frac{k}{x}\right)^{\alpha}$ $(x > k; \alpha > 0)$, середнє значення $E[X] = \frac{\alpha}{\alpha - 1}k$ $(\alpha > 1)$.

Такі специфічні характеристики властиві не тільки трафіку даних (протоколи *TCP, FTP*), а й сигнальному трафіку (протокол *SS7*), *VBR-відео*, *Ethernet/ISDN* і деяким іншим. Фізично вони пояснюються високим ступенем групування пакетів на клієнтських ділянках, у маршрутизаторах і вузлах комутації інформаційно-комунікаційних мереж. Навіть якщо джерело породжує регулярний потік пакетів, дані до споживача доставляються серіями, що перемежовуються інтервалами простою. Причинами цього є обмежена швидкість роботи мережних пристроїв, недостатній обсяг буферів і ін.

Крім того, самоподібний трафік має особливу структуру, що зберігається при багаторазовому масштабуванні, — у реалізації зазвичай наявна деяка кількість викидів за відносно невеликого середнього рівня трафіка (рис. 2.13). Через такі сплески навантаження характеристики мережі також погіршуються: збільшуються втрати, затримки, «джитер» («тремтіння» або порушення синхронізації) пакетів при проходженні через вузли мережі.

Методи розрахунку вимог до мереж нових поколінь (пропускної здатності каналів, ємності буферів і ін.), засновані на марківських моделях і формулах Ерланга, що з успіхом використовувалися при проектуванні телефонних мереж, можуть давати невиправдано оптимістичні рішення і призводити до недооцінки навантаження.



а) перші 33 хвилини

б) 27 хвилин, що залишилися

Рис. 2.13. Приклад реалізації високошвидкісного трафіка даних у локальній мережі *Ethernet*. Залежність довжин кадрів від часу

Для самоподібного трафіка результати класичної теорії масового обслуговування потрібно застосовувати з деякими застереженнями. З огляду на пульсуючий характер самоподібного трафіка, загалом не можна вважати потік заявок найпростішим, оскільки на інтервалі спостереження не виконується умова стаціонарності. Однак, виходячи з логіки надання послуг з гарантованою наскрізною якістю обслуговування *QoS*, потрібно вимагати забезпечення якості обслуговування на інтервалі будь-якої тривалості, випадково обраному з загального сеансу передавання даних. Як на інтервалах з низькою інтенсивністю трафіка, так і на інтервалах, де спостерігаються сплески навантаження, трафік можна з достатньої для практики точністю вважати локально-стаціонарним. Уся проблема полягає в тому, щоб визначити моменти переходу від одного інтервалу до іншого. У принципі, звичайно, можна запропонувати алгоритми адаптації до змін навантаження, наприклад, з оцінюванням кореляційних властивостей потоку даних, однак навряд чи варто очікувати прийнятної точності, а отже, й високої ефективності таких алгоритмів.

На наш погляд, більший інтерес становить одержання асимптотичних порівняльних оцінок для класичного пуассонівського і самоподібного потоків.

Розглянемо одноканальну систему масового обслуговування (СМО) з чеканням класу *GI / G / 1*. Оскільки кореляційна функція самоподібного трафіка не є експонентною, вхідний потік заявок варто вважати потоком з обмеженою післядією. Заявки надходять у послідовні дискретні моменти $t_i, t_{i+1}, \ldots, t_n, \ldots, \ t_j \le t_{j+1}$ для кожного j, інтервали між ними $\tau_n = t_n - t_{n-1}$ незалежні і розподілені за тим самим законом $F_n(\tau) = P\{\tau_n < \tau\}, \quad n \ge 2$.

Тривалості обслуговування заявок — незалежні величини ς_n із законом розподілу $\Psi_n(\varsigma) = P\{\varsigma_n < \varsigma\}, \quad n \ge 1$. Позначимо $\xi_n = \varsigma_{n-1} - \tau_n$. Тоді за умови, що послідовності $\{\tau_n\}$ і $\{\varsigma_n\}$ взаємно незалежні, можна визначити ймовірність

$$\Theta(\tau) = P\{\xi_n < \tau\} = \int_0^\infty \overline{F_n}(\eta - \tau)\, d\Psi_n(\eta), \qquad (2.24)$$

де $\overline{F_n}(\eta - \tau) = 1 - F_n(\eta - \tau)$.

Позначимо тривалість чекання n-ї заявки через ω_n. Якщо n-на заявка надійде одразу слідом за $(n-1)$-ю, їй, з урахуванням величини інтервалу τ_n, доведеться чекати обслуговування $\omega_{n-1} + \varsigma_{n-1} - \tau_n =$

$= \omega_{n-1} + \xi_n$ одиниць часу. Однак за досить великих τ_n величина $\omega_{n-1} + \xi_n$ може формально стати негативною. Ясно, що в цьому випадку дійсний час чекання n-ї заявки буде дорівнювати нулю — черги немає, і заявка надходить на обслуговування одразу з приходом. Отже, виконується рекурентне співвідношення

$$\omega_n = \max_n \{\omega_{n-1} + \xi_n, 0\}. \qquad (2.25)$$

Позначимо $G_n(x) = P\{\omega_n < x\}$. Тоді співвідношення (2.25) можна виразити через функції розподілу в такий спосіб:

$$G_{n+1}(x) = \begin{cases} \int_{-\infty}^{x} G_n(x-y)\, d\Theta(y), & x > 0, n \ge 2; \\ 0, & x \le 0, n \ge 1. \end{cases} \qquad (2.26)$$

Доповнимо вираз (2.26) очевидним співвідношенням для функції розподілу часу чекання першої заявки:

$$G_{n+1}(x) = \begin{cases} 1, & x > 0; \\ 0, & x \le 0. \end{cases} \qquad (2.27)$$

Вирази (2.24) і (2.26) являють собою інтеграли Стилтьєса, що у випадку безперервних розподілів $\Psi_n(t)$ і $\Theta_n(t)$ перетворюються в звичайні інтеграли. Таким чином, використовуючи вирази (2.26) і (2.27), можна рекурентно обчислювати розподіл тривалості чекання для заявки з будь-яким номером. Крім того, виявляється, що вони застосовні і при взаємній залежності послідовностей випадкових величин $\{\tau_n\}$ і $\{\varsigma_n\}$. Має значення лише незалежність величини ξ_n.

Логічно припустити, що час обслуговування заявки, наприклад час обробки пакета у вузлі комутації, не зв'язано твердою функціональною залежністю з довжиною пакета. Тоді, знаючи характеристики тривалості пакетів на вході комутатора як СМО, можна на ділянках локальної стаціонарності вхідного трафіка конкретизувати параметри розподілу часу обслуговування. Наприклад, при групуванні однорідних пакетів (що характерно для самоподібного трафіка) можна зробити припущення про детермінований час обслуговування (модель *GI / D / 1*).

З огляду на здобуті результати проаналізуємо вимоги до характеристик програмного комутатора для найпростіших і самоподібного вхідного потоків. За самоподібної природи трафіка залежність серед-

ньої тривалості черги (відповідно, необхідного розміру буфера) q від середнього коефіцієнта використання має такий вигляд:

$$q = \frac{\rho^{1/2(1-H)}}{(1-p)^{H/(1-H)}}. \qquad (2.28)$$

При $H = 0{,}5$ ця формула спрощується:

$$q = \rho/(1-\rho), \qquad (2.29)$$

що являє собою класичний результат СМО з найпростішим вхідним потоком і експоненційно розподіленим часом обслуговування ($M/M/1$). Для системи з детермінованим часом обслуговування ($M/D/1$) класичний результат виглядає так:

$$q = \frac{\rho}{1-\rho} - \frac{\rho^2}{2(1-\rho)}. \qquad (2.30)$$

Результати проведених за формулами (2.28)—(2.30) розрахунків зображено на рис. 2.14.

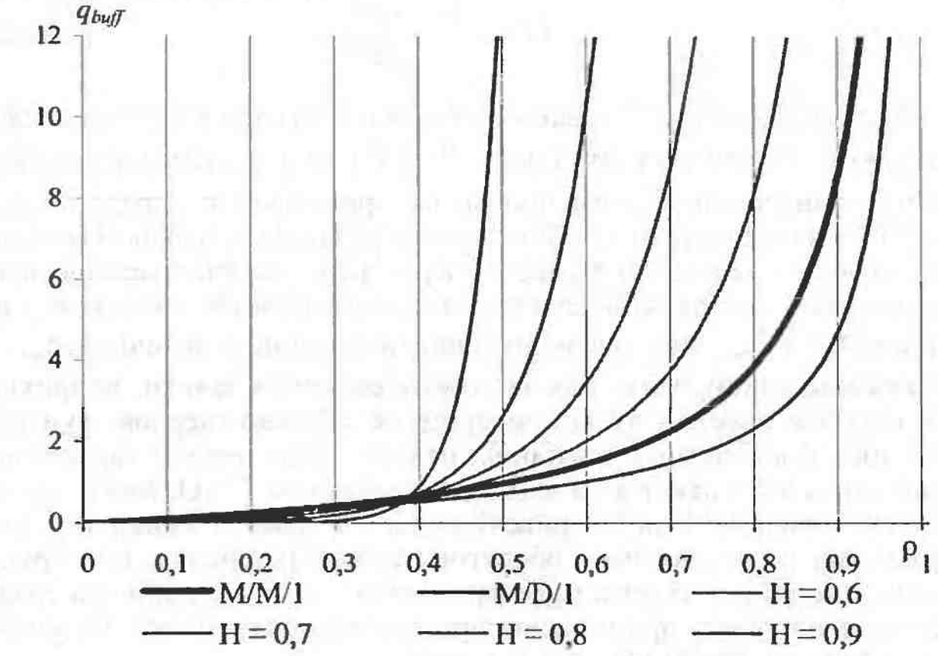


Рис. 2.14. Залежності довжини черги заявок (потрібної пам'яті буфера q_{buff}) від коефіцієнта використання ρ для різних моделей вхідного трафіка

На графіках добре видно, що для самоподібного трафіка вже при $\rho \approx 0{,}4$ потрібен більший ресурс пам'яті буферних пристроїв, аніж для класичної моделі $M/M/1$, що вважається найменш сприятливою порівняно з іншими (наприклад, з постійним або гаусовським розподіленим часом обслуговування). Швидкість росту необхідного обсягу пам'яті росте за збільшення параметра Херста, що зумовлено, в основному, ступенем групування однорідних пакетів і сплесками навантаження на мережу.

Можна також зробити висновок, що просте нарощування буферної пам'яті (апаратним або програмним способом) є малоефективним. При очікуваному збільшенні частки трафіка даних у загальному обсязі ступінь самоподібності буде збільшуватись, і залежність $\rho(q_{buff})$ дедалі різкішатиме. Позначена тенденція підвищення продуктивності комутаційного устаткування при розробці нових зразків програмних комутаторів *muny Softswitch* уселяє деякий оптимізм, однак не можна забувати, що будь-які мережні ресурси зненацька швидко виснажуються в разі безперервної появи нових послуг і прикладних програм. Гарні можливості зниження коефіцієнта використання виникають при збільшенні кількості незалежних рівнобіжних каналів програмного комутатора з організацією загальної черги до кількох вхідних портів. При цьому спрощуються алгоритми обробки пріоритетних потоків трафіка даних програмними методами. Тому важливим завданням є постійне удосконалювання програмного забезпечення вузлів комутації, зокрема, інтерфейсів прикладного програмування *API*.

Ми вже розглядали СМО з чеканням, обмеженим тільки довжиною черги. У такі СМО заявка, що надійшла в чергу, вже не полишає її і «терпляче» чекає обслуговування. Однак у мережах систем УПР весь трафік мережі з різнорідними потоками вже не може вважатись «еластичним», тобто не має обмежень на час чекання. Для таких ізохронних прикладних програм як передача голосу при перевищенні часу затримки більш ніж на 100... 150 мс різко знижується якість відтворення. Погіршується розбірливість мови, що в деяких ситуаціях може призвести до неприпустимих наслідків. Теоретично в разі перевищення порога чекання заявка може піти з черги (так звані *нетерплячі* заявки). Практично, звичайно, у системах УПР така ситуація неприпустима. Для СМО з «нетерплячими» заявками поняття «ймовірність відмови» не має сенсу — кожна заявка стає в чергу, але може і не дочекатись обслуговування, пішовши завчасно. Для запобігання втратам таких «нетерплячих» заявок і, відповідно, погіршення якості обслуговування доцільно застосовувати алгоритми обслуговування з пріоритетами типу маркірування потоків або ранжирування пакетів за тривалістю.

Необхідно врахувати також, що існує ненульова ймовірність вичерпання буферної пам'яті вузла комутації (ВК) — маршрутизатора, звичайного або програмного комутатора. Іншими словами, кількість місць у черзі не можна вважати необмеженою. При цьому заявкам, що знову надійшли, буде відмовлено в обслуговуванні. Імовірність відмови в обслуговуванні залежить від співвідношення інтенсивності надходження й обслуговування заявок. Не обслуговані заявки (пакети, кадри, повідомлення) просто губляться. Доведеться передавати їх повторно, оскільки джерело не одержить квитанцію — підтвердження про доставку.

Розглянемо характеристики навантаження на мережу за наявності «нетерплячих» заявок і обмеженому обсязі буферної пам'яті.

Якщо всі канали обслуговування зайняті і є черга заявок, то, як відомо, потік обслугованих заявок можна вважати найпростішим. Зробимо також допущення про найпростіший характер потоку «нетерплячих» заявок у загальному потоці. Відносна пропускна здатність системи q обчислюється з припущення, що будуть обслуговані всі заявки, крім тих, котрі підуть з черги достроково. Для знаходження середньої кількості таких заявок обчислимо середню кількість заявок у черзі:

$$\bar{r} = 1p_{n+1} + 2p_{n+2} + ... + rp_{n+r} + ...$$

На кожну з них діє «потік відходів» з інтенсивністю ν. Отже, із середньої кількості $\bar{r}$ заявок у черзі в середньому йтиме, не дочекавшись обслуговування, $\nu\bar{r}$ заявок в одиницю часу; усього в одиницю часу в середньому буде обслуговано $A = \lambda - \nu\bar{r}$ заявок.

Відносна пропускна здатність СМО буде $q_p = 1 - \frac{\nu}{\lambda}\bar{r}$, середня кількість зайнятих каналів (із загальної кількості n) $\bar{z} = \rho - \beta\bar{r}$, середня кількість заявок у черзі $\bar{r} = \frac{\rho}{\beta} - \frac{\bar{z}}{\beta}$. Тут λ, μ — інтенсивності потоку заявок і обслуговування відповідно; $\rho = \frac{\lambda}{\mu}$, $\beta = \frac{\nu}{\mu}$.

Припустимо, що всі канали обслуговування ВК мають спільну буферну пам'ять обсягом N_b осередків. Канали взаємонезалежні, дисципліна доступу до кожного каналу однакова (наприклад, FIFO—*first in—first out*, «перший прийшов—перший вийшов», LIFO—*last In—first out*, «останній прийшов—перший вийшов» або FIRO—*first In—random out*, «перший прийшов—випадковий вийшов»). Тоді можна розглядати будь-який канал ВК як одноканальну СМО з кількістю місць у черзі $N_Q = N_b/K$, де K — кількість каналів. Строго кажучи, N_Q є випадковою величиною, однак за умови $N_b >> K$ (що звичайно виконується на практиці) можна розглядати N_Q як деяке усереднене значення числа місць у черзі для кожного каналу.

Заявки обслуговуються за таких умов:

1. У сучасних ВК час обслуговування, як правило, не залежить від характеристик заявки (довжини пакета). Тому можна вважати, що інтенсивність обслуговування в кожному каналі однакова і постійна: $\mu_i = \mu = \text{const}$, $i = \overline{1,K}$.

2. На вхід ВК надходять як звичайні, так і «нетерплячі» заявки.

3. Час чекання «нетерплячої» заявки в черзі не перевищує t_{wmax}. Після цього заявка йде з черги.

4. Звичайна («терпляча») заявка відкидається, тобто йде з черги, якщо минув час чекання квитанції (тайм-ауту) t_{out}.

5. «Нетерпляча» заявка йде з черги і більше не з'являється на даному ВК, оскільки джерело — термінальний вузол або проміжний ВК — перемаршрутизує її. При цьому $t_{wmax} << t_{out}$. Дана умова випливає з такої вимоги. Необхідно повторно (а можливо й двічі) передати «нетерплячу» заявку по іншому маршруту, причому загальний час передавання не повинен перевищувати максимально припустимий час затримки для даного виду трафіка.

6. «Терпляча» заявка, в якої минув час тайм-ауту, може з'явитися на цьому самому ВК через деякий, загалом недетермінований час. Цей час залежить від затримок проходження заявки від джерела до розглянутого КК і є досить малим. Логічно припустити, що за цей час статистичні характеристики вхідного трафіка на даному ВК можуть змінитися досить незначно.

Стани каналу обслуговування можуть бути такими:

— S_0 — канал вільний з імовірністю P_0; у випадку надходження заявки її обслуговування починається негайно;

— S_1 — канал зайнятий, черги немає; ймовірність стану S_1 дорівнює P_1;

— S_2 — канал зайнятий, одна заявка в черзі; ймовірність стану S_2 дорівнює P_2;

.
.
.

— S_k — канал зайнятий, $k-1$ заявка в черзі; ймовірність стану S_k дорівнює P_k;

.
.
.

— S_{Nq+1} — канал зайнятий, N_Q заявок у черзі, вільних місць у черзі немає; ймовірність стану S_{Nq+1} дорівнює P_{Nq+1}.

Події $S_0, S_1, \ldots, S_{Nq+1}$ являють собою повну групу, отже, сума ймовірностей $P_0, P_1, \ldots, P_{Nq}$ дорівнює одиниці. Ймовірність того, що на якомусь конкретному часовому інтервалі всі місця в черзі зайняті звичайними і «нетерплячими» заявками, за умови, що «нетерплячі» заявки в даний момент не залишають черги, дорівнює $P_{\text{відмv}} = \dfrac{\lambda - \nu}{\lambda - \nu + \mu} = P_{Nq+1}$. Позначимо різницю між інтенсивністю вхідного потоку λ і потоку відходжень «нетерплячих» заявок ν через $\lambda_0 = \lambda - \nu$. Нагадаємо, що μ — інтенсивність обслуговування.

Для того, щоб не накладати обмеження на вид імовірнісних розподілів інтервалів між первинними і повторними заявками, скористаємося моделлю потоків з обмеженою післядією. Тоді можна записати вираз для $P_{\text{відмv}}$ у такому вигляді: $P_{\text{відмv}} = \dfrac{\lambda_0}{\lambda_0 + \mu}$. Ймовірність $P_{\text{відмv}}$, власне кажучи, являє собою частку не обслугованих заявок із загальної кількості вхідних заявок, як звичайних, так і «нетерплячих». Позначимо цю частку

$$q_{\text{необсл}} : P_{\text{відмv}} = \frac{\lambda_0}{\lambda_0 + \mu} = q_{\text{необсл}}.$$

Відповідно до наведених вище умов обслуговування звичайні заявки через якийсь час можуть повернутися для повторного обслуговування на ВК. Отже, вони додаються до нових заявок, що надійшли, й інтенсивність вхідного потоку зростає на величину $\left(1 + q_{\text{необсл}}\right)$. Нова величина інтенсивності вхідного потоку

$$\lambda_1 = \lambda_0\left(1 + q_{\text{необсл}}\right) = \lambda_0 + \frac{\lambda_0^2}{\lambda_0 + \mu}. \qquad (2.31)$$

Приблизно через такий самий час на вході ВК знову будуть присутні первинні заявки, до яких додадуться раніше не обслуговані. За

аналогією з (2.31) запишемо вираз для знову зміненої інтенсивності сумарного вхідного потоку:

$$\lambda_2 = \lambda_1\left(1 + q_{\text{необсл}}\right) = \lambda_1 + \frac{\lambda_1^2}{\lambda_1 + \mu}.$$

За індукцією запишемо вираз для поточної інтенсивності λ_i вхідного потоку заявок у такому вигляді:

$$\lambda_i = \lambda_{i-1} + \frac{\lambda_{i-1}^2}{\lambda_{i-1} + \mu}. \qquad (2.32)$$

У даній рекурентній послідовності (2.32) зміни інтенсивності вхідного потоку заявок ураховуються всі заявки, не обслуговані на попередніх етапах. При прямуванні $i \to \infty$ ми одержимо якісь асимптотичні оцінки корисної пропускної здатності мережі за різних співвідношень $\rho = \lambda/\mu$ — інтенсивності надходження заявок і інтенсивності їх обслуговування. На рис. 2.15 зображені графіки загальної і корисної пропускної здатності мережі для різних співвідношень між вихідною інтенсивністю вхідного потоку λ_0 й інтенсивністю обслуговування μ.

Розглянемо далі задачу оцінювання ймовірності виникнення колізії в мережі *Ethernet*. Оскільки локальні мережі систем УПР будуються, як правило, за технологією *Ethernet*, проблема обмеження коефіцієнта використання мережі є ще більш гострою з причин, пов'язаних саме з принципами роботи, закладеними в цій технології.

За збільшення коефіцієнта використання мережі $k_{\text{вик}}$ — відносини пропускної здатності до інтенсивності трафіка — дедалі більша частина ресурсу витрачається на обробку колізій. При прямуванні $k_{\text{вик}}$ до одиниці буде прямувати до одиниці ймовірність колізій і, відповідно, поява все нових і нових *jam*-послідовностей. Намагаючись обробляти колізії, мережа перестане пропускати корисну інформацію і працюватиме «на себе». При обґрунтуванні гранично припустимого коефіцієнта використання мережі необхідно враховувати ризик виникнення такої ситуації.

Нехай до мережі підключено N комп'ютерів, що видають потоки пакетів з інтенсивністю $\lambda_n(t)$, $n = \overline{1, N}$. Будемо вважати, що пакети, видавані n-м комп'ютером, мають тривалість τ_n. Щоб не зв'язувати себе необхідністю точного обліку моменту появи кожного пакета, припустимо, що вони відбуваються у випадкові моменти

часу з однаковим імовірнісним розподілом, і на кінцевому відрізку часу $T >> \tau_n$ утворять потік Ерланга k-го порядку.

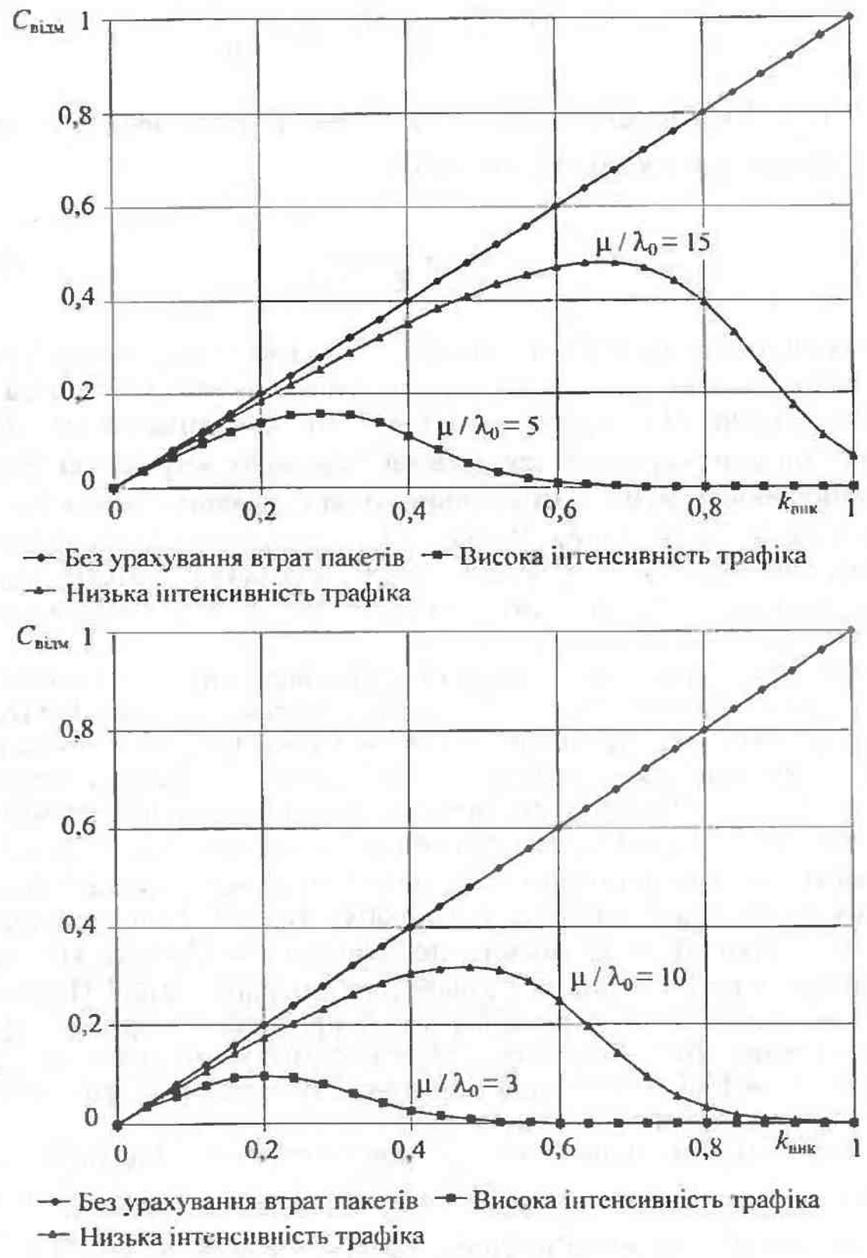


Рис. 2.15. Реальна пропускна здатність мережі при втратах і повторних передаваннях пакетів

Задаючися порядком потоку Ерланга, можна одержати будь-який ступінь післядії: від повної взаємної незалежності між моментами появи подій при $k = 0$ до детермінованого функціонального зв'язку при $k \to \infty$.

Нехай у точці t_j, що знаходиться усередині інтервалу аналізу T, з'явився пакет f_j із тривалістю τ_j. Імовірність появи цього пакета позначимо $P(t_j)$. Виведемо вираз для умовної щільності ймовірності часткового перекриття пакета f_j іншим пакетом f_i із тривалістю τ_i. Можливі положення пакетів зображено на рис. 2.16.

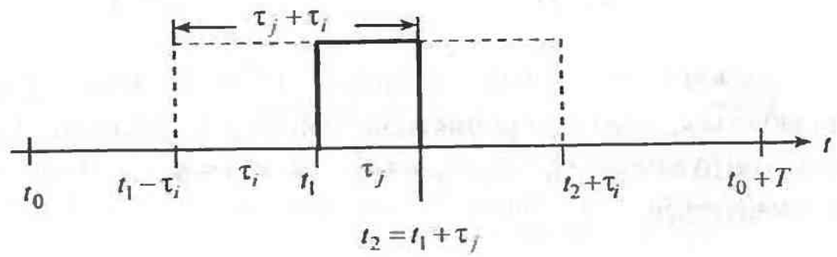


Рис. 2.16. Можливі положення пакетів у мережі *Ethernet*

Будемо вважати, що перекриття відбувається вже при зіткненні пакетів. Тоді необхідно обчислити спільну ймовірність $P(t_i, t_j)$ того, що пакет f_i потрапить в інтервал $[t_1 - \tau_i,\ t_1 + \tau_j]$ за умови, що пакет f_j починається в точці t_1. З урахуванням часових співвідношень (див. рис. 2.16) ця ймовірність визначається як

$$P(t_i, t_j) = P(t_1 - \tau_i \le t \le t_1 + \tau_j / t_1 = t_j).$$

Логічно припустити, що процеси появи в мережі пакетів, видаваних різними джерелами, взаємно незалежні, а моменти появи пакета f_j на інтервалі $[t_0, t_0 + T]$ рівноймовірні. Тоді

$$P(t_i, t_j) = P(t_1 - \tau_i \le t \le t_1 + \tau_j / t_1 = t_j) = \\ = P(t_1 - \tau_i \le t \le t_1 + \tau_j) P(t_1),$$

де $P(t_1) = P(0 \le t_j < t_j + \tau_j \le T) = \tau_j / T$ — імовірність того, що пакет f_j «накриє» інтервал тривалістю τ_j в якій-небудь точці t_j.

Припустімо, що в момент часу $t_1-\tau_i$ пакет f_i не спостерігається. Обчислимо ймовірність $P\left(t_1-\tau_i \le t_i \le t_1+\tau_j\right)$ появи f_i-го пакета за відрізок часу $\left[t_1-\tau_i, t_1+\tau_j\right]$ використовуючи вирази для закону Ерланга k-го порядку, у яких для обліку нестаціонарності введемо перемінну інтенсивність потоку $\lambda_i(t)$:

$$P\left(t_1-\tau_i \le t \le t_1+\tau_j\right)= \\ =\frac{\lambda_i(t_1-\tau_i)\left(\lambda_i(t_1-\tau_i+t)\,t\right)^{k-1}}{(k-1)!}\exp\left(-\int_{t_1-\tau_i}^{t_1+\tau_j}\lambda_i(t)\,dt\right). \quad (2.33)$$

Якщо вважати інтенсивність потоку $\lambda_i(t)$ величиною, що повільно знижується, можна з прийнятною точністю усереднити її на інтервалі інтегрування $\left[t_1-\tau_i, t_1+\tau_j\right]$: $\lambda_i(t)\approx\lambda_{i\text{сер}}$. Тоді вираз (2.33) спрощується:

$$P\left(t_1-\tau_i \le t \le t_1+\tau_j\right)=\frac{\lambda_{i\text{сер}}\left(\lambda_{i\text{сер}}\,t\right)^{k-1}}{(k-1)!}\exp\left(-\lambda_{i\text{сер}}\right). \quad (2.34)$$

Після нескладних, але громіздких перетворень вираз (2.34) перетвориться для випадку самоподібного трафіка за аналогією з виразом (2.28):

$$P\left(t_1-\tau_i \le t \le t_1+\tau_j\right)= \\ =\frac{\lambda_{i\text{сер}}^{1/2(1-H)}\left(\lambda_{i\text{сер}}^{1/2(1-H)}\,t\right)^{k-1}}{(k-1)!}\exp\left(-\lambda_{i\text{сер}}H/(1-H)\right), \quad (2.35)$$

де H — параметр Херста.

За формулами (2.33)—(2.35) розраховані залежності корисної пропускної здатності мережі від коефіцієнта використання (рис. 2.17) для пуассонівського трафіка і самоподібного трафіка з різними параметрами Херста.

Під ідеальним трафіком ми розуміємо відсутність колізій. Помітно, що продуктивність мережі спадає вже при коефіцієнті використання, що перевищує значення 0,3...0…0,4 (особливо для самоподібного трафіка).

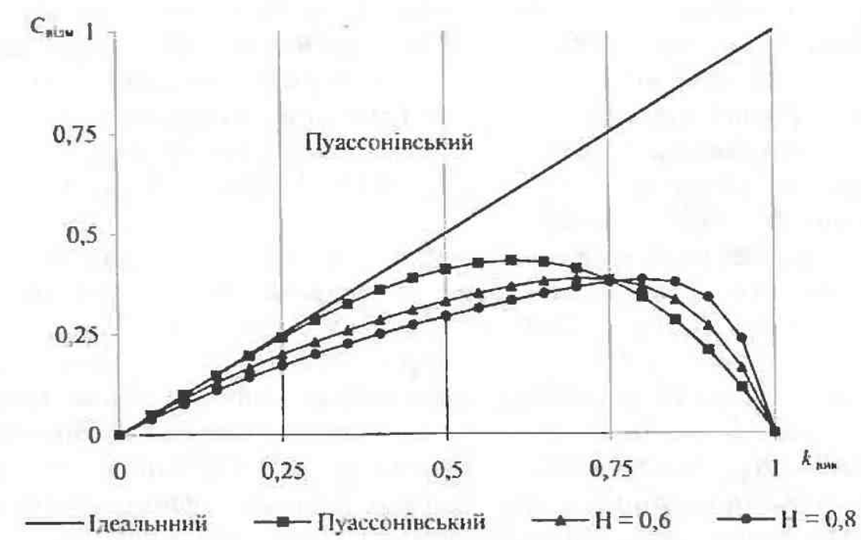


Рис. 2.17. Залежність відносної пропускної здатності мережі *Ethernet* від коефіцієнта використання при різних видах трафіка даних (*H* — параметр Херста самоподібного трафіка)

Таким чином, коефіцієнт використання мереж, побудованих як за АТМ-*технологією*, так і за технологією *Ethernet*, не може бути занадто близький до одиниці, інакше корисна пропускна здатність мережі різко впаде. Мережа буде або повторно передавати загублені пакети і квитанції, або обробляти колізії, тобто працювати «на себе». В жодному разі не можна допускати неконтрольоване перевантаження мережі, оскільки відновлення параметрів заявленої продуктивності мережі йде набагато повільніше, ніж їх падіння. Мережні фахівці говорять, що «мережа швидко лягає, але повільно встає». Такі ситуації тим більше неприпустимі для систем критичного застосування. Тому для керування ресурсами і підтримки пропускної здатності мережі в цілому на необхідному рівні необхідно перерозподіляти навантаження на окремі сегменти вже з появою найперших симптомів перевантаження, поки вона ще контрольована.

2.3.3. Метод адаптивної логічної структуризації локальної обчислювальної мережі

Логічна структуризація мереж будь-якого масштабу — один з основних способів подолання обмежень, що виникають при використанні загального поділюваного середовища. Великі мережі, як пра-

вило, будуються на основі структури з загальною магістраллю (*Backbone*). До магістралі через ВК і мережі доступу приєднуються автономні підмережі. Підмережі, своєю чергою, поділяються на сегменти, призначені для обслуговування структурних підрозділів, груп автоматизованих робочих місць (АРМ) або окремих АРМ, що входять до складу аеродромно-районних АС УПР (АРАС УПР) або районних АС УПР (РАС УПР).

Для сегментації магістральної мережі найбільш доцільно використовувати шлюзи, оскільки з їх допомогою успішно вирішуються завдання об'єднання підмереж із різними типами системного (мережного) і прикладного програмного забезпечення. Крім того, за допомогою шлюзів локалізується трафік окремих ділянок магістральної мережі. Перелічені завдання ще ефективніше і з більш високою швидкодією вирішуються за допомогою програмних комутаторів (*Softswitch*), однак ці пристрої значно дорожчі за шлюзи.

При сегментації мереж досягаються такі переваги:

1. Сегментовані мережі більш гнучко адаптуються до потреб окремих груп користувачів. У різних підмережах можуть використовуватися різні мережні технології, операційні системи і прикладне ПЗ. Це не виключає можливостей обміну даними між підмережами через ВК.

2. Поліпшується контроль ієрархії і розподілу прав доступу до мережних ресурсів.

3. Спрощується керування мережею в цілому. Загальна задача керування розпадається на сукупність локальних задач, більш простих і найчастіше слабко зв'язаних між собою. Тому при розв'язанні проблем усередині однієї підмережі умови роботи інших підмереж практично не змінюються.

Природно, структуризація і сегментація мереж не може бути доведена до виродженого стану, коли кожен термінальний вузол фактично являє собою окремий сегмент. Тоді кількість ВК буде практично дорівнювати кількості термінальних вузлів. Крім істотного подорожчання системи в цілому, це може призвести до негативних наслідків. Через затримки проходження даних (кадрів, пакетів) через ВК, втрату даних у разі переповнення буферів і вимушених повторів пересилання загублених даних пропускна здатність мережі може різко зменшитися. Крім того, будуть утрачені такі очевидні переваги мережі зі стандартною технологією поділюваного середовища:

— простота топології і легкість нарощування кількості термінальних вузлів у межах, що визначаються стандартом технології;

— простота регулювання потоку даних і доступу користувачів до загального поділюваного середовища (наприклад, за допомогою арбітра мережі);

— простота і повна уніфікація протоколів обміну, що забезпечує простоту конструкції, а отже, низьку вартість і високу надійність мережного устаткування.

Тому, мабуть, існує розумний оптимум у виборі кількості сегментів мережі і кількості термінальних вузлів у кожному сегменті. При цьому зовсім не обов'язково, щоб кожен сегмент мав однакову кількість вузлів — усе визначається інтенсивністю потоків даних в окремому сегменті.

Задачу оптимізації топологічної структури мережі поставимо в такий спосіб. Маємо вектори:

$\vec{U}$ — вектор параметрів мережного навантаження: інтенсивності потоків даних між кожною парою сусідніх вузлів комутації, статистичні характеристики трафіка, пріоритети або деякі вагові функції окремих потоків даних;

$\vec{Q}$ — вектор параметрів якості сервісу мережі: швидкодія, вірогідність й ін., що характеризують якість передачі даних або, у традиційному трактуванні – якість сервісу *QoS*;

$\vec{W}$ — вектор експлуатаційних характеристик мережі: пропускна здатність каналів передачі даних, швидкодія й обсяги буферної пам'яті вузлів комутації, надійність і час відновлення устаткування, вагові коефіцієнти, за допомогою яких даються порівняльні оцінки параметрів логічних зв`язків між вузлами мережі.

Накладаються (векторні) обмеження на граничні характеристики мережного устаткування, у тому числі на загальну вартість і структуру мережі:

$$C_i\left(\vec{U}, \vec{Q}, \vec{W}\right) \le C_{i\max}, \; i = \overline{1, N_c}, \tag{2.36}$$

де N — кількість елементів множин C_i і $C_{i\max}$.

Нарешті, накладаються обмеження на технічну архітектуру мережі, що випливають з умов фізичної і практичної реалізованості: гранично досяжна швидкість передавання даних, максимально припустимі відстані між вузлами, мінімально досяжні затримки в комутаційному устаткуванні, рівень взаємних перешкод і т.д. Позначимо множину цих обмежень через $R_{\max}$:

$$R\left(\vec{U}, \vec{Q}, \vec{W}\right) \in R_{\max}. \tag{2.37}$$

Потрібно знайти такий набір векторів $\left\{\vec{U}, \vec{Q}, \vec{W}\right\}$, що доставляв би екстремум функціоналові нормованої ефективності

$$\Psi_{\text{пс}}\left(\vec{U}, \vec{Q}, \vec{W}\right) \xrightarrow[\substack{\vec{U}=\vec{U}_{opt} \\ \vec{Q}=\vec{Q}_{opt} \\ \vec{W}=\vec{W}_{opt}}]{} \max \qquad (2.38)$$

при обмеженнях виду (2.36)—(2.37).

Загальний алгоритм (рис. 2.18) адаптивної логічної структуризації мережі являє собою таку послідовність кроків:

1. Уведення вихідних даних:

— кількість активних термінальних вузлів N_i і їхня належність до i-го сегмента;

— усереднені обсяги завантаження буферів ВК;

— умови роботи системи (штатний режим — режим обробки екстремальних ситуацій; тип екстремальної ситуації ЕС_i);

— початкове співвідношення $V_{\text{вн}i}/V_{\text{мс}i}$ кожного сегмента;

— середній час тайм-ауту t_{out} при міжмережному обміні (в екстремальній ситуації менше ніж у штатної у 2—3 рази);

— середній коефіцієнт використання кожного сегмента $k_{\text{вик}i}$.

2. Чи склалася екстремальна ситуація (ЕС)?

Якщо так, то перехід на крок 3. Інакше — перехід на крок 6.

3. Розпізнавання типу ЕС.

4. Вибір із БД оптимальної схеми сегмента для обробки даної ЕС.

5. Перехід на крок 2.

6. Вибір із БД схеми початкової сегментації.

7. Підрахунок кількості *jam*-послідовностей у кожному сегменті.

8. Підрахунок кількості виявлених перевищень постійної складової у фізичному каналі кожного сегмента.

9. Розрахунок корисної пропускної здатності.

10. Розрахунок $k_{\text{вик}i}$.

11. $k_{\text{вик}i} < k_{\text{вик max}}$?

Якщо так, то перехід до кроку 2. Інакше — до кроку 12.

12. Пошук екстремуму — найкращої структури за критерієм $k_{\text{вик}i} \le k_{\text{вик max}}$.

13. Розрахунок нової структури i-го сегмента.

14. Перехід до кроку 2.

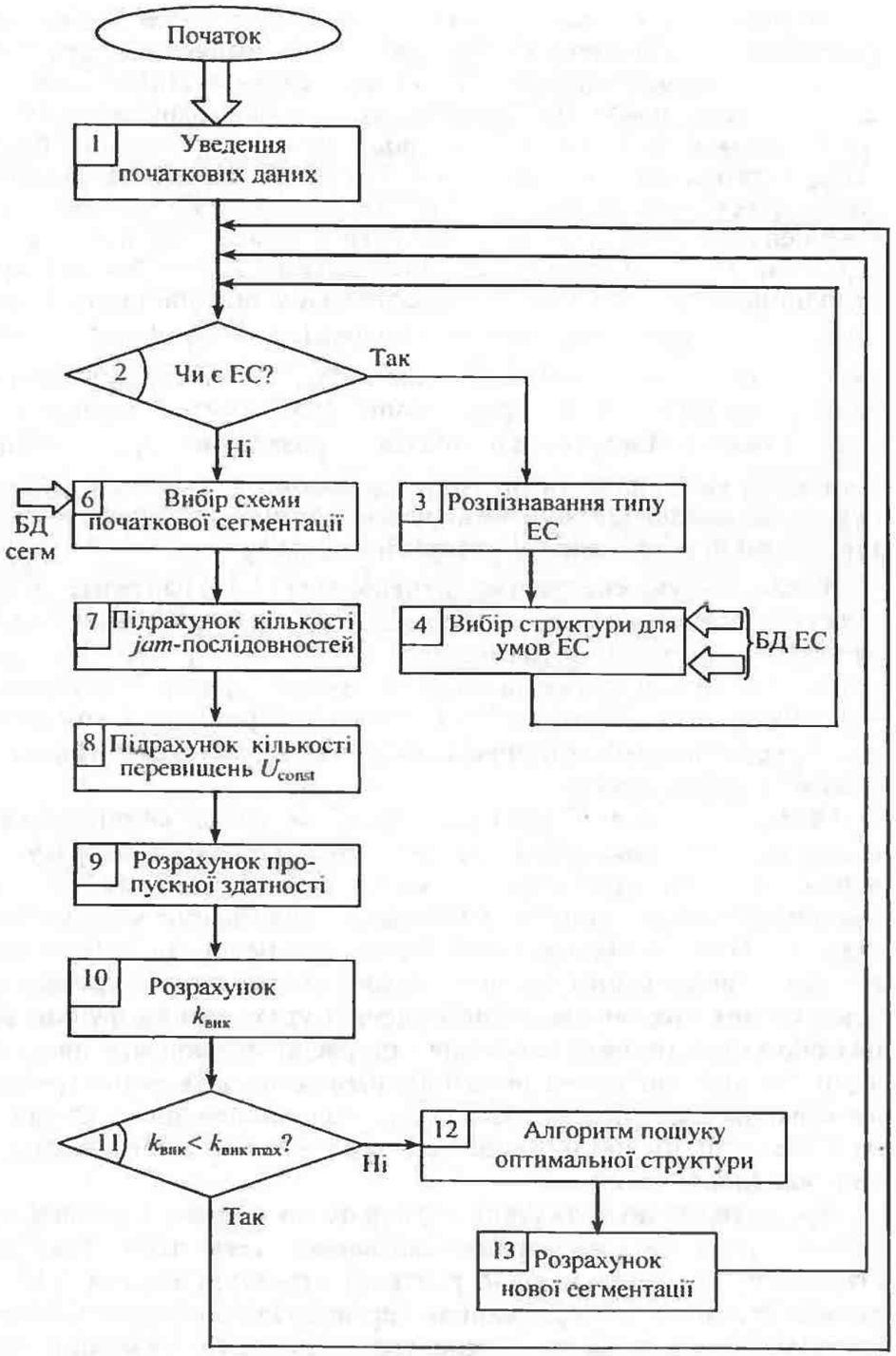


Рис. 2.18. Схема алгоритму адаптивної структуризації

Як видно з рисунка, алгоритм є циклічним з незаданим числом повторень. Для пошуку екстремуму — оптимальної структури — доцільно використовувати методи рекурентного статистичного пошуку. У базах даних (БД) зберігаються конфігурації сегментів, що організуються за наявності тієї або іншої екстремальної ситуації (ЕС), а також вихідна конфігурація мережі. Наявність перевантаження фіксується за кількістю *jam*-послідовностей і за кількістю перевищень постійної складової напруги в загальному поділюваному середовищі U_{const}. Кількість *jam*-послідовностей не зв'язана твердою функціональною залежністю з коефіцієнтом використання i-го сегмента $k_{\text{вик}\,i}$, тому його можна табулювати по графіках, подібних тим, що наведені на рис. 2.15, і вибирати дані з БД з урахуванням раніше накопичених апріорних даних про характер трафіка в кожному сегменті. Найпростіший підхід — розрахунок $k_{\text{вик}\,i}$ за співвідношенням між кількістю *jam*-послідовностей і кількістю корисних переданих кадрів. Останнє визначається за кількістю переданих квитанцій про прийом кадру j-м термінальним вузлом.

Задача пошуку екстремуму функціонала (2.37) належить до класу задач системного аналізу, як і будь-яка задача теорії великих систем. Критерій оптимізації функціонала (2.37) є векторним. Він складається з часткрвих критеріїв, причому багато критеріїв суперечливі. Рішення задачі (2.37) може бути отримане через пошук компромісів між суперечливими критеріями (наприклад, методами Парето або аналізу ієрархій Сааті).

Обмеження виду (2.35) і почасти (2.36) важко формалізувати й описати з використанням строгого математичного апарату. Тим більш важко, практично неможливо вирішити задачу (2.37) у цілому, намагаючись залишитися в рамках формального математичного підходу. Тому необхідно використовувати певні процедури декомпозиції — розкладання загальної задачі на сукупність окремих задач і пошук для них оптимальних рішень з урахуванням функціональних або статистичних (кореляція і регресія) зв'язків між цими задачами. Як такі процедури звичайно використовують метод групового врахування аргументів (МГВА), запропонований академіком А. Г. Івахненком, або метод діакоптики — послідовного аналізу частин складної системи.

Представимо досліджувану корпоративну мережу у вигляді ієрархічної структури «магістраль—підмережі—сегменти». Таке представлення цілком відповідає реальній структурі систем УПР, що завжди будуються за ієрархічною деревоподібною схемою. Можна з достатньою впевненістю стверджувати, що при оптимізації, наприклад, окремого сегмента мережі характеристики інших сегментів як

мінімум не погіршаться, а характеристики всієї мережі покращаться. Тому, залишаючись у рамках глобальної проблеми (2.38) оптимального проектування корпоративної мережі для системи критичного застосування, розглянемо окрему задачу оптимізації окремого її сегмента – локальної обчислювальної мережі структурного підрозділу АС УПР. Оскільки розмірність задачі може змінюватися в широких межах, використовуємо алгоритм рекурентного статистичного пошуку.

1. На основі $K+1$ випадкових незалежних дослідів маємо векторну суму

$$\vec{W}_m = \sum_{k=1}^{K} \Xi_k \left[\Psi_{\text{нс}}\left(\vec{W} + r\Xi_k\right) - \Psi_{\text{нс}}\left(\vec{W}\right) \right], \qquad (2.39)$$

де r — довжина випадкового кроку; Ξ_k — випадкова величина R, що має рівномірний розподіл на $[0, 1]$; K — кількість вузлів комутації. Нагадаємо, що $\vec{W}$ — вектор експлуатаційних характеристик мережі, компонентами якого в даному випадку є коефіцієнт використання і корисна пропускна здатність сегмента.

2. Суму (2.39) представимо в рекурентній формі:

$$\vec{W}_m = a\vec{W}_{m-1} + \Xi\Delta\Psi_{\text{нс}}, \quad 0 \le a \le 1,$$

де a — коефіцієнт убування післядії. Цей коефіцієнт вибирається дослідним шляхом для конкретного сегмента мережі з урахуванням середнього навантаження в штатній ситуації.

Для прискорення пошуку екстремуму застосуємо типовий алгоритм одномірної мінімізації уздовж обраного напряму. Як відомо, найбільш ефективними алгоритмами одномірного пошуку є метод Фібоначі і метод золотого перерізу. Оскільки число обчислень функції, необхідне для досягнення потрібного ступеня дроблення інтервалу невизначеності, заздалегідь невідомо, використовуємо метод золотого перерізу, що менш чутливий до недоліку апріорних зведень, ніж метод Фібоначі.

Розглянемо структуру алгоритму локальної одномірної мінімізації уздовж напряму статистичного градієнта. Нехай є N термінальних вузлів локальної мережі (рис. 2.19). Між термінальними вузлами є ВК, приєднані до кореневого ВК (КР ВК) для даного сегмента. ВК можуть працювати як у звичайному, так і в «прозорому» режимі. У звичайному режимі роботи ВК будуються таблиці маршрутизації й окремі сегменти ізолюються один від одного, аж до переходу в режим «мікросегментації», коли кожен термінальний вузол приєднаний до КР ВК через свій ВК. У «прозорому» режимі ВК працює

як простий повторювач, без переадресації пакетів, що надходять. Формується загальне поділюване середовище, для якого затримки передавання даних мінімальні, алгоритм обміну даними найбільш простий, а отже, надійність роботи сегмента найвища.

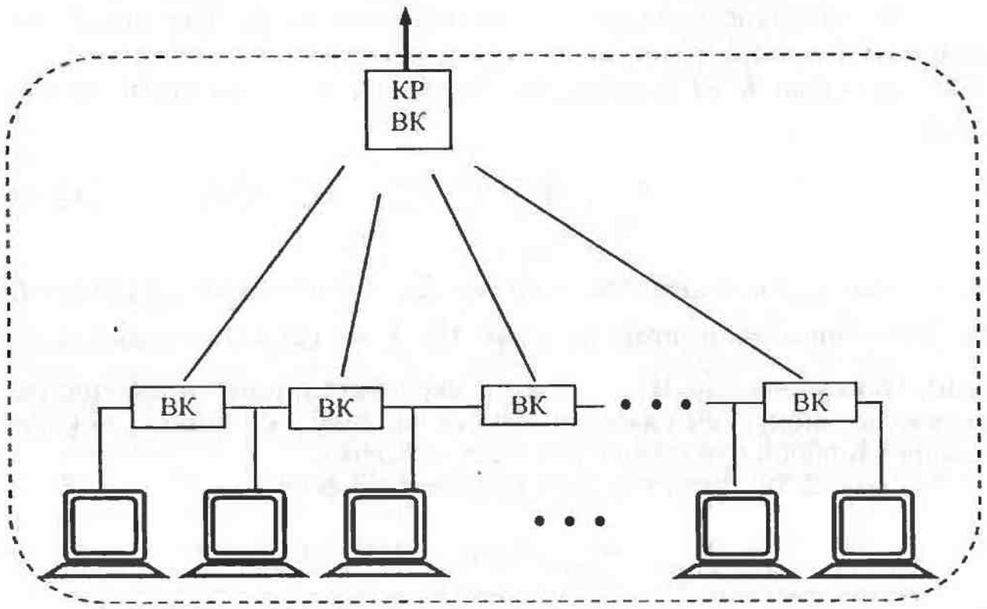


Рис. 2.19. Структурна схема автономної локальної мережі

У вихідному стані, за відсутності ознак перевантаження сегмента мережі, усі ВК працюють у «прозорому» режимі. Дані через них передаються транзитом. Сегмент являє собою загальне поділюване середовище й один домен колізій.

Алгоритм пошуку точки розбивки сегмента на підсегменти працює в такий спосіб:

1. При виявленні перевантаження сегмента розбиваємо його на дві частини $N_1 + N_2 = N$ за методом золотого перерізу в співвідношенні $\frac{N_1}{N} \approx \frac{N_2}{N_1}$.

2. Відшукуємо значення $\Psi_{не1}$ для N_1 і $\Psi_{не2}$ для N_2.

3. За здобутим значенням $\Psi_{не1}$, $\Psi_{не2}$ скорочуємо інтервал невизначеності $\min(N_1, N_2)$.

4. Представимо величину інтервалу невизначеності у вигляді $N_{k-2} \approx N_{k-1} + N_k$, $1 < k < K$.

5. Коефіцієнт дроблення інтервалу невизначеності на k-му етапі $\Delta N_k \approx N \times 0{,}618^{k-1}$. Якщо інтервал невизначеності на черговому етапі дроблення стає менше одиниці (правильним дробом), операція пошуку точки розбивки закінчується.

Алгоритм одномірної мінімізації (пункти 1—5) реалізується на кроці 12 загального алгоритму (рис. 2.18) і включається за командою на кроці 11. Така структура алгоритму дає можливість швидкої адаптації до змінюваної обстановки, перерозподілу мережних ресурсів, зосередження їх на розв'язанні позаштатних ситуацій (наприклад, конфліктів повітряних суден).

2.3.4. Рекомендації з вибору вигляду і структури інформаційно-обчислювальної підсистеми

Насамперед систематизуємо вимоги до обчислювальних пристроїв і комп'ютерних мереж, за допомогою яких забезпечується робота автоматизованих систем управління критичного застосування (АСУ КЗ). Зазначимо, що обсяг вимог до АСУ КЗ ширший, ніж просто до систем реального часу (СРЧ). Основною відмінністю АСУ КЗ від СРЧ є вимога високої надійності і живучості. Тобто маються на увазі не тільки порівняно великий середній час безвідмовної роботи і збереження своїх функціональних можливостей в екстремальних умовах, зокрема, в разі впливу техногенних, природних і людських чинників. Для АС УПР як системи критичного застосування найважливішою умовою є збереження працездатності (з відповідним обмеженням обсягу і якості розв'язуваних задач) при повних або часткових відмовах елементів, вузлів або навіть підсистем, що входять до її складу. Крім того, час відновлення працездатності в повному обсязі має бути мінімальним. Цей час визначається вимогами безперервності обслуговування повітряного руху: інтервали видачі команд і обміну даними звичайно становлять одиниці або навіть частки секунд. За цей час система має перейти на резервний пристрій або підсистему.

Природно, й устаткування, й програмне забезпечення інформаційно-обчислювальної підсистеми — стандартні комп'ютери або сервери, спеціалізовані обчислювачі, мережне устаткування і лінійно-кабельне обладнання, операційні системи і бази даних, спеціалізовані прикладні програми тощо — повинні працювати в реальному часі. Швидкодія інформаційно-обчислювальної підсистеми має бути такою, щоб вимога обслуговування АС УПР у реальному часі вико-

нувалася за найінтенсивнішого навантаження. У даному випадку мається на увазі максимальна очікувана інтенсивність повітряного руху для певного аеровузла або аеродрому. Необхідно враховувати ту обставину, що обчислювальні і комунікаційні ресурси устаткування, як уже зазначалося, зненацька швидко виснажуються через стрімке зростання обсягів нових послуг і прикладних програм. Отже, впроваджуючи нові комп'ютеризовані системи, необхідно заздалегідь закладати резерви обчислювальних і комунікаційних потужностей і передбачати можливості розширення і нарощування наявних систем.

Таким чином, основними вимогами до інформаційно-обчислювальної підсистеми АС УПР як системи критичного застосування є висока надійність і живучість, ремонтопридатність і поновлюваність. Оскільки АС УПР має при цьому працювати в реальному часі, першочергова вимога до інформаційно-обчислювальної підсистеми полягає у забезпеченні необхідної продуктивності за будь-яких запланованих, зокрема пікових, навантажень.

Звичайно, найпростішим і очевидним є розв'язання завдання «у лоб»: для підвищення надійності використовувати дорогі високонадійні вузли й елементи, резервувати цілі пристрої і лінії устаткування; для досягнення необхідної продуктивності — експлуатувати систему на граничних режимах. Зрозуміло, що при цьому одні результати іноді будуть досягатися за рахунок інших. Тому потрібно використовувати більш тонкі інструменти управління надійністю і якістю системи. У даному підрозділі зупинимося докладніше на мережних елементах інформаційно-обчислювальної підсистеми, оскільки вони є найбільш «вузьким місцем», і саме від них залежить результативна продуктивність.

Відповідно до прийнятих нині класифікацій обчислювальних (комп'ютерних) мереж, інформаційно-обчислювальну підсистему АС УПР можна віднести до корпоративних мереж.

Корпоративні мережі мають такі особливості:

1. За допомогою цих мереж покривається велика територія — країна або навіть континент. Територіальне рознесення вузлів мережі може досягати сотень і тисяч кілометрів.

2. Кількість користувачів (працівників станцій) може вимірятися тисячами, кількість серверів і спеціалізованих обчислювачів — сотнями, мейнфреймів — десятками.

3. Віддалені вузли зазвичай являють собою високошвидкісні локальні мережі. Ці мережі можуть бути зв'язані між собою через будь-які фізичні канали, а топологія, як правило, змішана, причому не виключена наявність петель.

4. Канали глобального зв'язку між локальними «острівцями» корпоративної мережі, як правило, повинні мати швидкодію такого самого порядку, що і швидкодія цих локальних мереж-острівців.

5. Процеси адміністрування корпоративних мереж, обліку користувачів, встановлення ієрархії і розподілу прав доступу повинні бути, по-перше, максимально автоматизовані, а по-друге, мати раціональний ступінь децентралізації.

Тут перелічені лише деякі, найбільш важливі особливості корпоративних мереж. Можна стверджувати, що корпоративна мережа АС УПР — це гетерогенна мережа зі змінною структурою. Змінність структури в даному випадку слід розуміти як випадкову послідовність змін у процесі функціонування мережі. Це зумовлено, по-перше, безперервною зміною кількості активних користувачів, по-друге, випадковими і найчастіше неконтрольованими змінами параметрів і структури мережі (наприклад, відмова устаткування), по-третє, істотною неоднорідністю трафіка даних, що циркулюють у мережі.

Під гетерогенністю мереж звичайно розуміють різнорідність комутаційного устаткування і каналотвірної апаратури, великі розходження (іноді на кілька порядків) у пропускній здатності окремих фрагментів мережі, наявність різних мережних технологій, протоколів обміну даними і т.д.

З огляду на сказане можна стверджувати, що корпоративна мережа автоматизованої системи керування повітряним рухом об'єктивно являє собою сегментовану структуру. Окремі сегменти функціонують досить самостійно і слабко впливають один на одного. Співвідношення обсягів внутрісегментного $V_{вн}$ і міжсегментного $V_{мс}$ трафіка $q_{tr} = V_{вн} / V_{мс}$ змінюється в широких межах. За результатами аналізу літератури загального і спеціального призначення можна зробити такі висновки:

1. У мережах загального призначення — комп'ютерних, телекомунікаційних, конвергованих — традиційне співвідношення $q_{tr} = 80 / 20$, тобто 80 % трафіка — звертання до локальних ресурсів усередині сегмента і 20 % трафіка — обмін даними між сегментами. Сьогодні воно трансформується убік збільшення частки міжсегментного обміну: «50 / 50» і навіть «20 / 80».

2. У мережах систем критичного застосування розглянуті співвідношення дуже сильно залежать від режиму роботи. У штатному режимі може мати місце співвідношення «80 / 20» і навіть менше.

В екстремальних ситуаціях воно змінюється до «20 / 80», причому пріоритет міжсегментного обміну є значно вищим. Фактично це веде до подальшого збільшення частки міжсегментного трафіка як більш пріоритетного: $q_{tr} = V_{вн} / V_{мс} \approx$ «10 / 90» і більше. Однак і менш

пріоритетний внутрісегментний трафік повинен обслуговуватися в реальному часі, особливо в екстремальних ситуаціях.

Таким чином, корпоративна інформаційно-обчислювальна мережа АС УПР має будуватися за «острівним» принципом. Топологія мережі, як правило, змішана. У ній можуть бути такі елементи:

— коміркова топологія;
— загальна шина;
— ієрархічна зірка;
— кільце.

Для досягнення необхідної продуктивності, сумісності мережних технологій і протоколів обміну, масштабованості і розширюваності мережі доцільно використовувати підхід, заснований на еталонній моделі взаємодії відкритих систем (*OSI — Open System Interconnection*). Модель *OSI* — це міжнародний стандарт, прийнятий спільно Міжнародною організацією зі стандартизації *ISO* і Міжнародним союзом з телекомунікації (сектор стандартів у сфері телекомунікації) *ITU-T*. Усі сучасні комп'ютерні і телекомунікаційні мережі будуються на базі моделі *OSI*. Детальний опис стандарту моделі *OSI* — це понад 1000 сторінок тексту. Для нас же важливі такі принципові особливості моделі *OSI*:

- семирівнева структура організації обміну даними: від нижнього — фізичного рівня, до верхнього — рівня прикладних програм;
- усі операції з обміну даними контролюються «зверху вниз»: якщо один із нижніх рівнів не виконав свої функції (або виконав їх з помилками), то на верхніх рівнях виниклі проблеми усуваються, а в разі неможливості їх вирішення операції нижніх рівнів повторюються.

У рамках моделі *OSI* побудова корпоративної мережі АС УПР є найбільш ефективною і наочною, оскільки полегшуються завдання перетворення інформації в процесі обміну. Уявімо, що кожен рівень обслуговується найближчим нижнім рівнем (є клієнтом нижнього рівня), а сам, своєю чергою, обслуговує найближчий верхній рівень (є, відповідно, сервером для верхнього рівня). Тоді ми дістанемо багатошарову (багаторівневу) модель мережі АС УПР, з багаторазовим використанням технології «сервер—сервер—клієнт—сервер». Структуру багатошарової моделі корпоративної мережі АС УПР зображено на рис. 2.20.

За такої організації мережі контроль, керування і модернізація децентралізовані. Усі ці процедури спрощуються, а ефективність їх виконання і надійність системи в цілому підвищуються.

У рамках запропонованої багатошарової моделі неважко логічно й технічно обгрунтувати структуру корпоративної мережі АС УПР. Як відомо, у цифрових конвергованих мережах (або мережах нових

поколінь — *NGN*) найпоширеніші — *ATM* і *IP*-технології. Основною перевагою протоколу *IP* є його простота і можливість динамічної фрагментації пакетів. Однак протокол *IP*, будучи, по суті, дейтаграмним протоколом, не дає жодних гарантій доставки повідомлень. Якщо при цьому на якій-небудь ділянці мережі сталася втрата пакетів, вузли комутації (або маршрутизатори) починають посилати запити своїм сусідам. Навантаження росте лавиноподібно і може взагалі паралізувати даний фрагмент мережі, що для умов критичного застосування неприпустимо.

З іншого боку, *ATM*-технологія гарна тим, що є високошвидкісною (швидкості до 622 Мбіт / с) і забезпечує універсальну обробку різнорідного трафіка в гетерогенній мережі. Крім того, *ATM*-технологія забезпечує гарантоване значення *QoS — quality of service* (якість сервісу).

Тому цілком логічне створення багаторівневої цифрової архітектури мережі виду *IP/ATM/SDH/DWDM*, де *IP — Internet Protocol* (протокол Інтернет);

ATM — Asynchronous Transfer Mode (технологія асинхронного режиму передавання або перенесення пакетів стандартного розміру 53 байта); *SDH — Synchronous Digital Hierarchy* (синхронна цифрова ієрархія); *DWDM — Dense Wave Division Multiplexing* – оптична технологія високощільного мультиплексування з поділом за довжиною хвилі.

З огляду на наведені міркування можна представити структуру корпоративної мережі АС УПР у вигляді магістральної мережі (так званої *backbone* або *core network*), до якої через мережі доступу підключаються багатосегментні мережі окремих аеровузлів/районів. Як відомо, така структура застосовується практично в будь-яких великих, у тому числі і глобальних, мережах завдяки своїй простоті і гарним можливостям нарощування і модернізації. Магістральна мережа являє собою *ATM*-мережу на основі оптоволокна. Мережі доступу і локальні обчислювальні мережі (ЛОМ) доцільно будувати на мідних кабелях типу витої пари (*Fast Ethernet*) або на оптичних кабелях (*Gigabit Ethernet*). Зазначимо, що оптичні технології в цілому набагато дорожчі, ніж технології на основі мідних кабелів (в основному через дорожнечу оптичного комутаційного устаткування і буферної пам'яті). Тому використовувати такі високошвидкісні підмережі доцільно тільки на найвідповідальніших ділянках, наприклад, у мережах управління, обробки конфліктних ситуацій і ін.

Судячи зі схеми 2.20, основним термінальним устаткуванням інформаційно-обчислювальної структури є автоматизовані робочі місця (АРМ) диспетчерів і керівників. Відповідно до концепції Гармо-

нізації національних систем організації повітряного руху держав — членів Співдружності Незалежних Держав, що базується, своєю чергою, на концепції аеронавігаційної системи «Євроконтролю», планується масове впровадження АРМ задля зниження завантаження операторів АС УПР. Оператори (диспетчери, керівники) вивільняються від рутинних, одноманітних, багаторазово повторюваних операцій і дістають більше можливостей для ефективного і швидкого розв'язання нестандартних завдань.

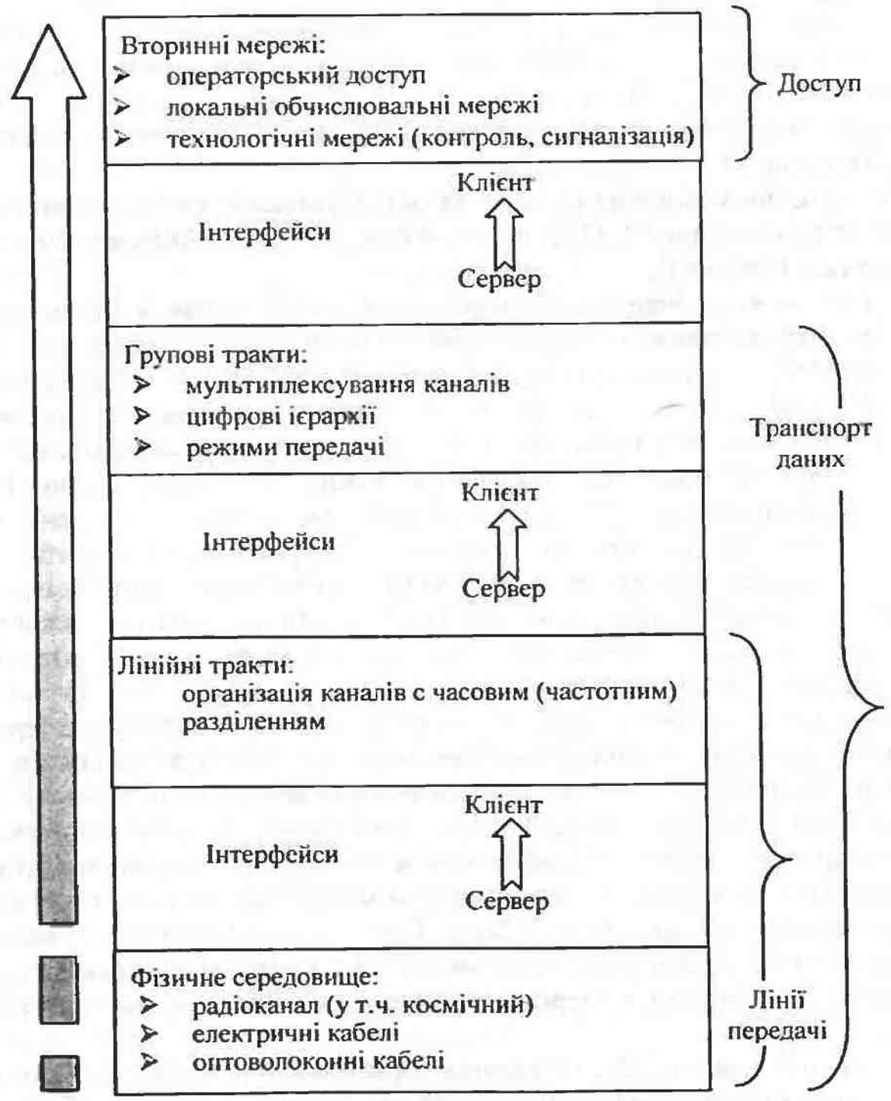


Рис. 2.20. Багатошарова модель корпоративної мережі АС УПР

АРМ різного призначення зв'язуються між собою через локальну обчислювальну мережу, виконану за однією з базових технологій ЛОМ. За результатами аналізу вітчизняних і закордонних розробок АС УПР можна дійти висновку, що найбільш застосовуваною базовою технологією ЛВМ є *Ethernet* різних модифікацій: стандартний 10 Мбіт, *Fast Ethernet, Gigabit Ethernet*.

Основною перевагою технології *Ethernet* є простота організації і керування ЛОМ. Зберігається наступність принципів організації попередніх і подальших модифікацій *Ethernet*. Тому модифікація мережного устаткування, нарощування мережі до меж, обумовлених стандартами *IEEE 802.x*, здійснюються досить просто.

У корпоративній мережі АС УПР також превалює «острівний» принцип, що досить поширений у телекомунікаційних і комп'ютерних мережах. Тут цей принцип об'єктивно є найпридатнішим внаслідок великого територіального рознесення локальних елементів корпоративної мережі, причому найчастіше — уздовж трас польотів (рис. 2.21).

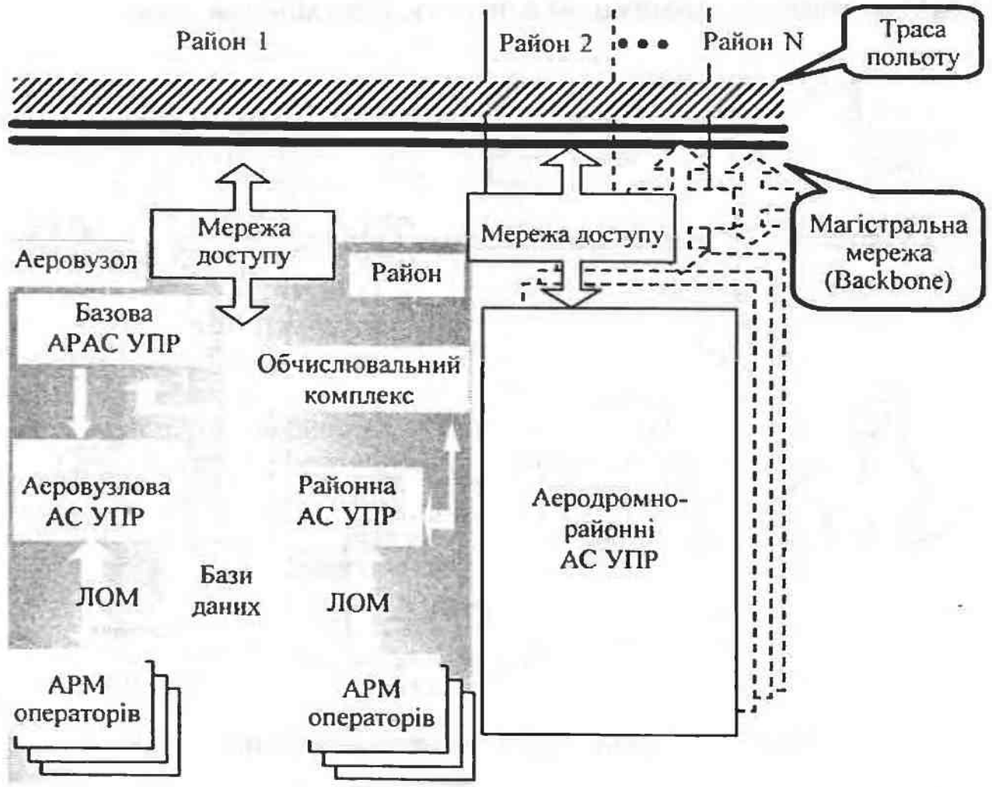


Рис. 2.21. Ієрархічна структура АРАС УП

Розглянемо один з типових «острівців» корпоративної мережі АС УПР — локальну обчислювальну мережу аеродромно-районної АС УПР (АРАС УПР) (див. рис. 2.21).

Вона містить універсальні і спеціалізовані обчислювачі, бази і сховища даних, набір АРМ операторів — диспетчерів, керівників, груп зв'язку, метео, довідкової інформації й ін.

Між окремими АРМ, що виконують самостійні функції, необхідні розв'язки по внутрішньому і міжсегментному трафіку. Також необхідні розв'язки і між обчислювальними мережами сусідніх районів. Розв'язки — це основа структуризації мереж із загальним поділюваним середовищем, що обслуговують системи реального часу й особливо системи критичного застосування. З огляду на наведені міркування і на основі результатів попередніх досліджень розроблена структурна схема обчислювальної мережі і набору АРМ диспетчерів АРАС УПР. На рис. 2.22 зображено варіант типової структури. Тут абревіатури *NA* (*network adaptor*) — мережний адаптер, *Hub* — концентратор, *Root hub* — кореневий концентратор. ТКНЗ означає «технологічний контроль, навігація і зв'язок».

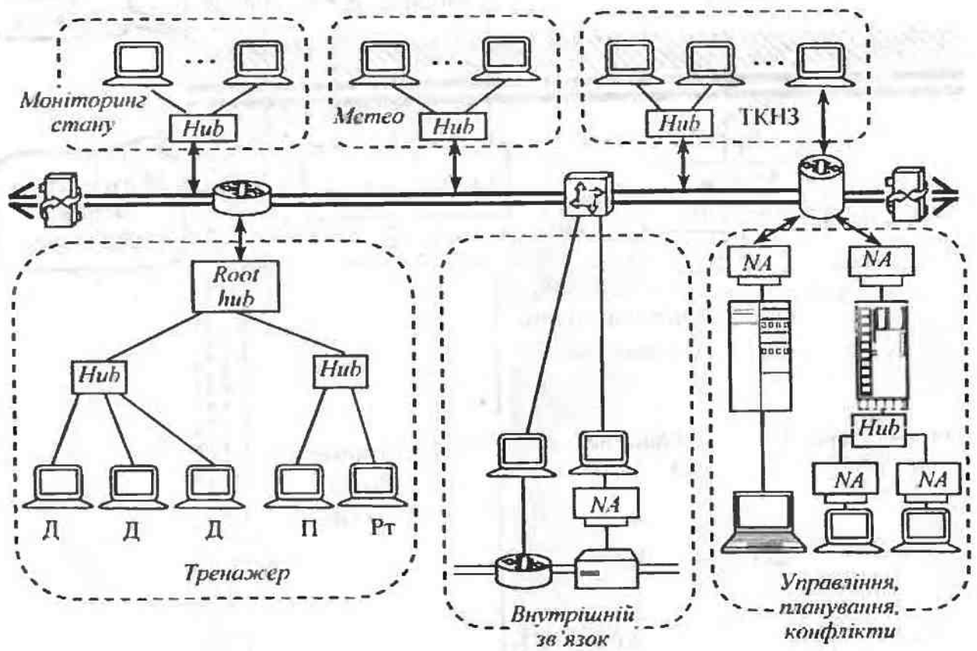


Рис. 2.22. Структурна схема типової мережі АС УПР

Мається на увазі зв'язок із зовнішніми абонентами (земля, повітряні судна, супутникові системи). Для внутрішнього зв'язку використовується окрема локальна мультисервісна мережа, по якій здійснюється транспорт як аналогових сигналів (термінальний вузол приєднується через модем), так і цифрових сигналів (через комутатор).

Передбачається, що локалізація ділянки магістральної мережі, що проходить через розглянутий район, здійснюється за допомогою шлюзів. Локалізація окремих ЛОМ, що обслуговують відповідні АРМ, може здійснюватися за допомогою комутаторів, а для найвідповідальніших ланок — за допомогою програмних комутаторів або маршрутизаторів з різними можливостями і наборами надаваних послуг. Такий вибір вузлів комутації ґрунтується на результатах порівняльного аналізу їхніх характеристик: ефективності, вартості, розширюваності або масштабованості мережі.

Для вирішення обчислювальних задач, зв'язаних з обробкою конфліктних або екстремальних ситуацій, сплесками інтенсивності повітряного руху тощо, до складу ЛОМ включаються спеціалізовані обчислювачі реального часу. Обробка статистики навантаження на магістральну і локальні обчислювальні мережі може здійснюватися в універсальному обчислювачі. Там же реалізуються алгоритми адаптивної логічної структуризації мереж.

Кількість термінальних вузлів кожного АРМ залежить від обсягу і напруженості розв'язуваних задач і визначається індивідуально для кожної конкретної АРАС УПР. Відповідно, і при загальному виборі конкретного мережного комутаційного, термінального і лінійно-кабельного устаткування необхідно враховувати безліч організаційних, технічних і економічних чинників.

Розроблені принципи побудови розглянутої інформаційно-обчислювальної підсистеми досить універсальні. Завдяки широким можливостям і простоті зміни режимів роботи комутаційного устаткування забезпечується швидка й ефективна адаптація логічної структури обчислювальних мереж підсистеми.

Таким чином, запропоновані структури корпоративної мережі і локальних обчислювальних мереж для систем критичного застосування (див. рис. 2.21, 2.22) можуть служити основою для побудови інформаційно-обчислювальної підсистеми АС УПР.

Питання для самоперевірки

1. Які числові характеристики використовуються для оцінки продуктивності мереж загального призначення?

2. З яких міркувань задається пропускна здатність мережі системи критичного застосування?

3. Чому має забезпечуватися мультисервісність в інформаційно-комунікаційній системі?

4. Якими принциповими особливостями характеризуються мережі нових поколінь?

5. Перелічіть параметри та статистичні характеристики трафіка конвергентних мереж.

6. Які будуть наслідки у разі перерозподілу видів навантаження на телекомунікаційні мережі?

7. Які розподіли з «важкими хвостами» використовують для опису щільностей імовірностей самоподібних потоків?

8. Виконайте аналіз вимог до характеристик програмного комутатора для найпростіших і самоподібного вхідних потоків.

9. Як змінюється швидкість зростання необхідного обсягу пам'яті зі збільшенням параметра Херста? Обґрунтуйте.

10. Назвіть характеристики навантаження на мережу за наявності «нетерплячих» заявок і обмеженого обсягу буферної пам'яті.

11. Як здійснюється оцінювання ймовірності виникнення колізії в мережі *Enternet*?

12. Для чого потрібна логічна структуризація мережі?

13. Що доцільно використовувати для сегментації магістральної мережі? Які переваги досягаються при сегментації мережі?

14. Наведіть послідовність дій загального алгоритму адаптивної логічної структуризації мережі.

15. Які характерні особливості мають корпоративні мережі?

16. За яким принципом має будуватися корпоративна інформаційно-обчислювальна мережа АС УПР? Обґрунтуйте.

17. Наведіть характеристику локальної обчислювальної мережі аеродромно-районної АС УПР.

2.4. Інструменти, прилади та засоби технічного обслуговування комп'ютерних систем і мереж

Для пошуку несправностей і ремонту КС на професійному рівні треба застосовувати спеціальні інструментальні засоби, які дають змогу ефективно виявляти ці несправності і швидко їх усувати. До таких засобів належать:

— діагностичні пристрої і програми для тестування компонентів комп'ютера;

— простий набір інструментів для розбирання і збирання;

— прилади для вимірювання напруги й опору, такі як цифровий мультиметр, логічні пробники і генератори одиночних імпульсів для перевірки цифрових схем;

— хімічні препарати (розчин для протирання контактів), пульверизатор з охолодною рідиною і балончик зі стиснутим газом (повітрям) для чищення деталей комп'ютера;

— набір тампонів для протирання контактів;

— спеціалізовані підручні інструменти (наприклад, інструменти, необхідні для заміни мікросхем (чіпів));

— тестові рознімання для перевірки послідовних і паралельних портів;

— прилади тестування пам'яті для оцінювання якості функціонування відповідних модулів;

— устаткування для тестування живлення комп'ютера на кшталт перемінних перетворювачів напруги (трансформаторів) і тестерів, які дають змогу перевірити ефективність використання живлення.

У деяких випадках може знадобитися комплект інструментів для паяння.

2.4.1. Підручні інструменти

Інструменти, необхідні для сервісного обслуговування майже всіх комп'ютерів, відносно прості. Більшість з них цілком може уміститися у невеликій сумці. Навіть інструменти вищого класу для професіонального застосування уміщаються у валізках.

Зазвичай комплект складається з набору інструментів, призначених для робіт з розбирання й збирання блоків комп'ютерів, до якого входять:

- гайковий ключ на 3 / 16 дюйма;
- гайковий ключ на 1 / 4 дюйма;
- маленька хрестоподібна викрутка;
- маленька плоска (звичайна) викрутка;
- середня хрестоподібна викрутка;
- середня плоска (звичайна) викрутка;
- пристрій для витягування мікросхем із гнізд;
- пристрій для встановлення мікросхем у гнізда;
- пінцет;
- затискач для деталей;
- викрутки T10 і T15 типу *Torx*.

Деякі інструменти з наведеного списку практично не використовуються, однак вони завжди мають бути у наборі.

Гайкові ключі застосовуються для гвинтів із шестигранними голівками, якими в більшості комп'ютерів кріпляться кришка системного блоку, плати адаптерів, дисководи, блоки живлення і гучномовці. Гайковим ключем за такого кріплення користуватися зручніше, ніж звичайною викруткою.

Оскільки деякі виробники замість гвинтів із шестигранними голівками застосовують звичайні або хрестоподібні гвинти, то достатньо й викрутки.

Виходячи з вищесказаного слід використовувати універсальний набір викруток і ключів, приклад якого наведено на рис. 2.23.

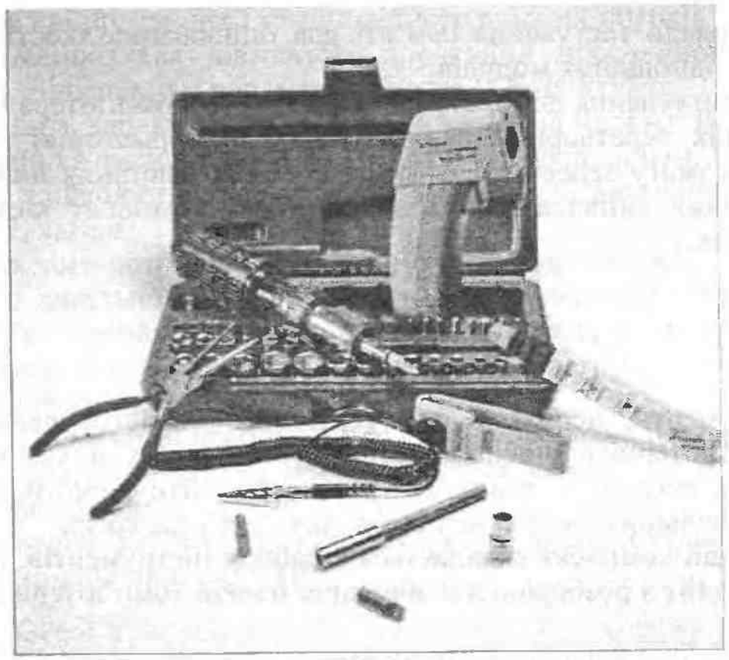


Рис. 2.23. Універсальний набір інструменту

Для роботи усередині корпуса комп'ютера дуже зручні інструменти з намагніченими кінцями. За допомогою таких інструментів досить просто установити і закрутити гвинт у важкодоступному місці або ж вийняти кріпильний елемент, що впав у «глибини» комп'ютера. Треба виявляти особливу обережність під час роботи з такими інструментами, оскільки деякі елементи комп'ютера (наприклад, жорсткі диски) чутливі навіть до дуже слабких магнітних полів.

Пристрої для виймання із гнізд й установлення мікросхем потрібні для того, щоб у процесі цих операцій не зіпсувати їхні виводи, а також панель.

У сучасних системних платах використовуються рознімання типу ZIF (*Zero Insertion Force* — з нульовим зусиллям вставляння). При вставлянні мікросхем у такі рознімання (і їх вийманні) зовсім не потрібно докладати зусиль, що дає змогу легко заміняти мікросхеми. Це насамперед стосується мікросхем процесора, а також модулів пам'яті. Єдиною мікросхемою у сучасних системних платах, що не оснащується такими розніманнями, є мікросхема *BIOS*. Сучасні мікросхеми з *BIOS* доволі специфічні і встановлюються у відповідні панелі (рис. 2.24), що зумовлено необхідністю виймання їх лише в разі виходу з ладу. Проте така необхідність існує, а некваліфіковані дії в разі їх заміни (особливо при вийманні) можуть призвести до потреби заміни панелі. Тому для заміни цих мікросхем слід використовувати спеціальний прилад (рис. 2.25).

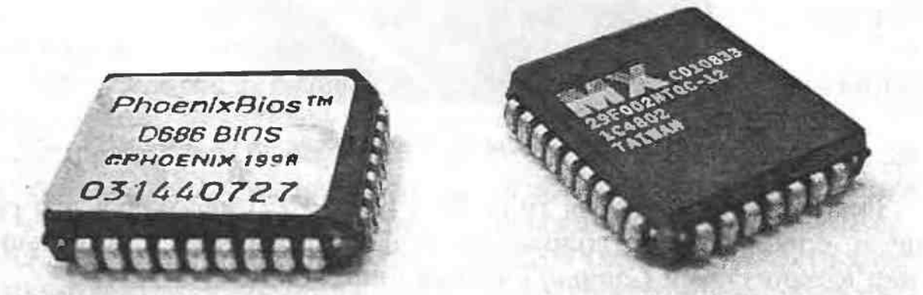
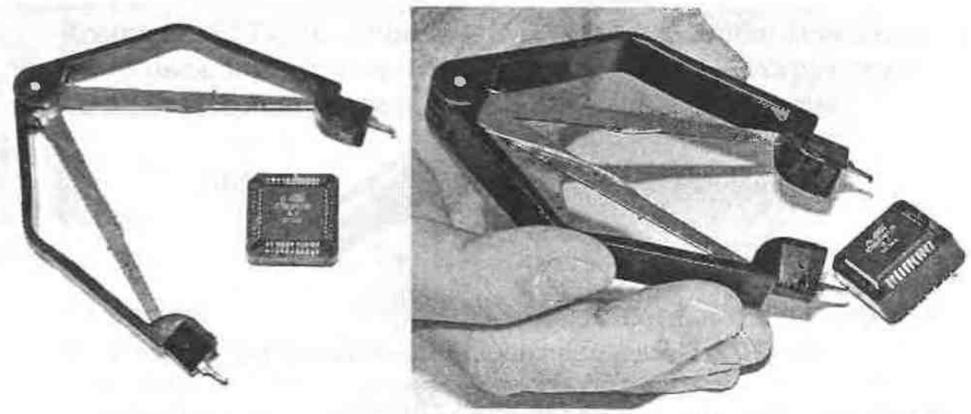


Рис. 2.24. Мікросхеми *PLCC—BIOS*



а *б*

Рис. 2.25. Прилад для виймання *PLCC*-мікросхем:
а — зовнішній вигляд приладу; *б* — спосіб застосування

Пінцетом або затискачем тримають невеликі гвинти або перемички, що незручно брати рукою (рис. 2.26). Вони особливо зручні для виймання невеликих деталей зсередини комп'ютера. За їх допомогою це можна зробити, не розбираючи його.

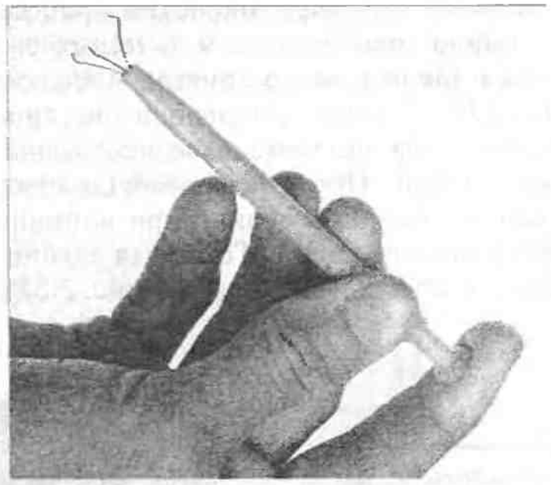


Рис. 2.26. Затискач для тримання невеликих деталей

Зіркоподібна викрутка типу *Torx* (рис. 2.27) необхідна для гвинтів зі спеціальними голівками, що застосовуються в більшості комп'ютерів фірми *Compaq* і деяких інших фірм.

Рис. 2.27. Насадка зіркоподібної викрутки типу *Torx*

Крім того, бажано до переліченого інструменту додати такі:
— пасатижі з довгими губками;
— лещата або затискач;
— пристрій для різання і зачищення проводів;

— метричні гайкові ключі;
— напилок;
— невеликий ліхтарик.

Пасатижами можна випрямляти виводи мікросхем, знімати і встановлювати перемички, монтувати кабелі і рознімання, а також тримати невеликі деталі.

За допомогою лещат можна монтувати рознімання і згинати кабелі для надання їм потрібної форми; лещата потрібні і для закріплення деталей при виконанні деяких операцій.

Затискачі корисні для захвачування маленьких компонентів на кшталт перемичок.

Пристроєм для різання і зачищення проводів користуватися набагато зручніше, ніж скальпелем або ножем. Він застосовується для обробки кабелів і проводів.

Метричні гайкові ключі можуть придатися для роботи з комп'ютерами європейських й азійських виробників, а також з PS/2 з метричним кріпленням.

Напилок може знадобитися для обробки гострих країв корпуса і шасі, а також для припасування лицьових панелей дисководів.

Ліхтарик призначений для освітлення комп'ютера всередині, особливо важкодоступних місць, коли звичайної лампи недостатньо. На мій погляд, ліхтарик — один із найважливіших інструментів.

Обов'язково придбайте комплект ESD (комплект електростатичного розвантаження) для захисту від електростатичних розрядів. Він складається з браслета з заземлювальним проводом і провідного килима з заземленням. Такий комплект убереже мікросхеми від ушкодження випадковою статичною електрикою.

Комплект ESD, як і інші інструментальні засоби, можна придбати в торговельних фірмах. Маючи всі перелічені інструменти і пристрої, можете починати ремонт або збирання комп'ютера.

2.4.2. Кріпильні деталі

2.4.2.1. Типи кріпильних деталей

Працюючи з комп'ютером ви можете натрапити на безліч різноманітних кріпильних деталей.

У більшості комп'ютерів застосовуються гвинти із шестигранною голівкою, для яких придатні гайкові ключі на 1/4 і 3/16 дюйма. Фірма IBM застосовує такі гвинти у своїх PC; вони ж використовуються в більшості сумісних комп'ютерів. Однак можливе застосування й інших кріпильних деталей. Наприклад, фірма *Compaq* у біль-

шості своїх комп'ютерів використовує гвинти типу *Torx* (вони мають зіркоподібний проріз у голівці). Викрутки різних розмірів для цих гвинтів позначаються так: T-8, T-9, T-10, T-15, T-20, T-25, T-30, T-40 і т.д.

Різновидом гвинтів *Torx* є секретні гвинти, що застосовуються в блоках живлення і деяких вузлах: вони схожі на звичайні гвинти цього типу, але в центрі прорізу в них є штир. Для них потрібна спеціальна викрутка з поглибленням під цей штир (звичайними інструментами викрутити такий гвинт неможливо). Єдиний спосіб зробити це без відповідної викрутки — обережно зрізати штир невеликим зубилом. Як правило, за допомогою таких гвинтів збираються вузли, що не розраховані на розкриття і заміняються цілком.

Багато виробників застосовують більш поширені стандартні гвинти, призначені для хрестоподібних і плоских викруток. Звичайно, працювати з такими гвинтами простіше, але вони менш надійні ніж шестигранні гвинти і гвинти *Torx*, оскільки їхній проріз під шліц досить легко зірвати. Дуже дешеві гвинти кришаться під викруткою, і крихти металу можуть потрапити на системну плату. Не створюйте собі зайвих проблем і намагайтеся не користуватися такими гвинтами.

2.4.2.2. Дюймова і метрична міри

Кріпильні деталі комп'ютерів можуть бути двох типів — дюймовими і метричними. Фірма IBM у більшості своїх комп'ютерів застосовує дюймове кріплення, але багато виробників користуються метричними гвинтами і гайками.

Американські моделі зроблені в дюймовому стандарті, а японські і тайванські — у метричному.

Якщо ви заміняєте накопичувач на гнучких дисках у старій моделі PC, то можете стикнутися з цією проблемою. Найчастіше на це натрапляють при заміні дисководів. Намагайтеся разом з обраним комп'ютером одразу придбати необхідні гвинти і кронштейни, оскільки знайти їх окремо в магазинах буде нелегко. В інструкції з експлуатації завжди наводяться креслення розташування отворів для кріплення і типи використаних гвинтів.

Накопичувачі на жорстких дисках можуть бути зроблені і в тому, і в іншому стандарті залежно від фірми-виробника. Сьогодні виробники більшості типів накопичувачів в основному використовують метричний стандарт.

Деякі гвинти (особливо для кріплення накопичувачів на жорстких дисках) повинні мати строго встановлену довжину. Занадто довгий монтажний гвинт, затягнутий до кінця, може пошкодити корпус

накопичувача. Перш ніж остаточно встановлювати новий диск, треба обережно спробувати закрутити гвинти і визначити, на яку глибину їх можна вкрутити без ризику зачепити корпус або інші частини накопичувача. Якщо у вас виникають сумніви, зазирніть у документацію — у ній точно сказано, які гвинти необхідні для кріплення.

2.4.3. Вимірювальні прилади

Іноді при перевірці плат або компонентів доводиться користуватися вимірювальними приладами і спеціальними пристроями. Вони порівняно недорогі і прості в застосуванні. Для перевірки комп'ютера необхідні мультиметр і тест-рознімання. Тест-рознімання дають змогу перевіряти послідовні і паралельні порти і кабелі, що приєднуються до них. Мультиметром можна вимірювати різні параметри, наприклад напругу в різних точках схеми або на виході блоку живлення, і перевіряти на обрив провідник на платі або кабель. Непоганим доповненням може бути тестер електричної розетки, за допомогою якого перевіряють правильність підімкнення електропроводки до розетки живлення, а також тестер мережної розетки й ліній з'єднання (телефонних, витої пари, USB і коаксіального кабеля).

2.4.3.1. Тест-рознімання

Для перевірки послідовних і паралельних портів застосовуються тест-рознімання (рис. 2.28). Якщо встановити їх замість з'єднувальних кабелів, то при перевірці будуть подаватися сигнали з вихідних контактів послідовних або паралельних портів на вхідні контакти, тобто на самих себе.

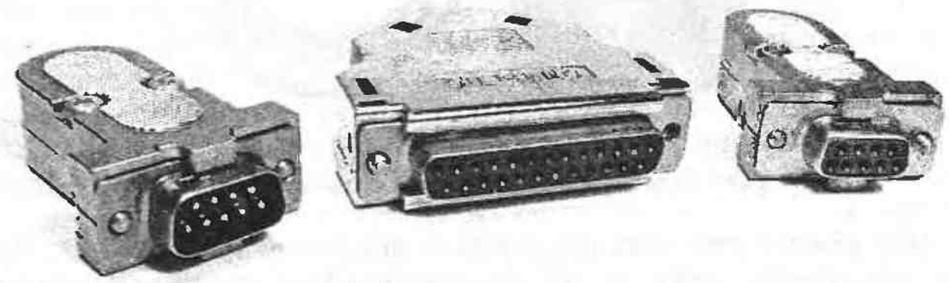


Рис. 2.28. Зовнішній вигляд типових 9- і 25-контактних тест-рознімань

Існує кілька типів тест-рознімань. Вам будуть потрібні рознімання для 9- і 25-контактних послідовних портів і для 25-контактного паралельного порту. Такі тест-рознімання випускаються багатьма фірмами, зокрема IBM (причому вона пропонує й універсальне рознімання, у якому всі три види об'єднані в одному корпусі).

2.4.3.2. Мультиметри

Найчастіше в процесі роботи доводиться вимірювати напругу й опір. Для цього застосовуються цифрові або аналогові мультиметри. У кожного з них є мінімум два вимірювальні виводи (щупи), котрі підмикаються до ланцюга, що перевіряється. При з'єднанні мультиметр відображає показання. Залежно від обраного режиму роботи прилад вимірює або опір, або постійну, або перемінну напругу (більш висококласні моделі можуть вимірювати струм, ємність, частоту, параметри транзисторів і т.п.). На рис. 2.29 наведено багатофункціональний мультиметр DT9207.

Рис. 2.29. Багатофункціональний мультиметр DT9207

Для кожної величини існує кілька діапазонів вимірювання. Наприклад, верхні межі шкали при вимірюванні постійної напруги можуть бути 200 мв, 2, 20, 200 і 1000 В. Оскільки в комп'ютерах використовується напруга живлення +5 і +12 В, найкраще виконувати вимірювання у діапазоні 20 В. На нижчих діапазонах прилад зашкалюватиме або він узагалі вийде з ладу, а на великих точність зчитування показань буде недостатньою.

Якщо величина вимірюваної напруги заздалегідь не відома, треба установити мультиметр на «найгрубший» діапазон, а потім поступово збільшувати чутливість. У кращих з цих приладів вибір діапазону вимірювання здійснюється автоматично. Такі мультиметри досить прості у використанні. Досить перемикнути мультиметр у режим вимірювання тієї величини, що необхідно, наприклад у режим постійної напруги, і приєднати щупи до ланцюга, котрий перевіряється. Мультиметр сам вибере оптимальний діапазон вимірювання, й операторові залишиться лише зчитати показання. Подібні прилади переважно цифрові.

Стрілочні вимірювальні прилади можуть становити небезпеку для цифрових схем, оскільки при вимірюванні опору на щупи подається від батареї тестова напруга зі значеннями, що перевищують максимально припустимі. У цифрових приладах ця напруга, як правило, становить 3—5 В.

2.4.3.3. Логічні пробники та генератори одиночних імпульсів

При пошуку несправностей у цифрових схемах зручно використовувати логічний пробник (рис. 2.30). Цифровий сигнал може бути або високого (5 В), або низького (0 В) рівня. Імпульси бувають дуже короткими (частки мікросекунди), а частота їхнього проходження може досягати десятків мегагерц, тому звичайний мультиметр у такій ситуації марний. Логічний пробник призначений для контролю й індикації саме таких цифрових сигналів.

Особливо він може придатися для пошуку поломки в «мертвому» комп'ютері. За допомогою пробника можна перевірити роботу тактового генератора і наявність інших синхронізуючих сигналів. Порівнянням сигналів на кожному виводі будь-якої інтегральної схеми із сигналами на неушкодженій мікросхемі можна знаходити компоненти, що вийшли з ладу. Логічний пробник може виявитися корисним і для перевірки дисководів — він дає змогу перевірити сигнали на інтерфейсному кабелі або в самій схемі накопичувача.

Разом із логічним пробником звичайно використовується генератор одиночних імпульсів. Він призначений для примусової подачі в схему імпульсу високого рівня (+5 В) тривалістю 1,5 — 10 мкс. Реакція схеми порівнюється з її «штатним» поводженням. Генератор одиночних імпульсів використовується рідше ніж логічний пробник, але в деяких випадках він буває досить корисний.

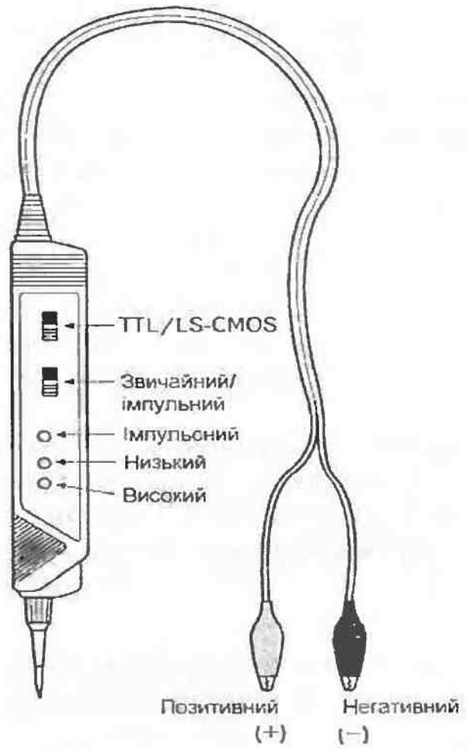


Рис. 2.30. Зовнішній вигляд логічного пробника

2.4.3.4. Тестер електричної розетки

Тестер електричної розетки — досить корисний вимірювальний пристрій. Цей простий і дешевий прилад застосовується для перевірки електричних розеток. Його вставляють у розетку і за світінням трьох світлодіодів визначають правильність підімкнення проводів.

Правильно змонтована мережна розетка — на жаль, велика рідкість. У більшості випадків у розетках неправильно підведений провідник заземлення. Неправильно змонтована розетка призводить до порушень в роботі комп'ютера і врешті решт до його «зависання». Це спричинено тим, що перешкоди мережі живлення у разі незаземленої системи потрапляють на загальний провід комп'ютера, відносно якого «зчитуються» рівні логічних сигналів. У результаті виникають помилки під час передачі даних і періодичні збої.

Іншою ознакою поганого заземлення електричних розеток є електричні розряди, що виникають у момент дотику до корпуса комп'ютера. У цьому випадові струм протікає не там, де потрібно. Використовуючи простий тестер електричних розеток, можна швидко визначити їхній стан.

2.4.3.5. Тестер мережних розеток і кабелів

Якість функціонування комп'ютерної системи в цілому часто залежить насамперед від якості мережі, за допомогою якої проводиться передача окремих даних, масивів інформації тощо. Виходячи з цього тестування лінійного мережного обладнання є необхідною складовою забезпечення ефективної роботи мережі КС.

Для тестування мережних ліній та приладів з'єднання (розеток) існує велика кількість приладів, які різняться універсальністю призначення, конструктивними особливостями, точністю вимірювання параметрів, складністю і багатьма іншими показниками. Як приклад можна навести недорогий універсальний тестер для тестування ліній з витої пари, телефонного зв'язку, USB і коаксіального кабелю, а також з'єднань типу RJ-45, RJ-11, USB, BNC (рис. 2.31).

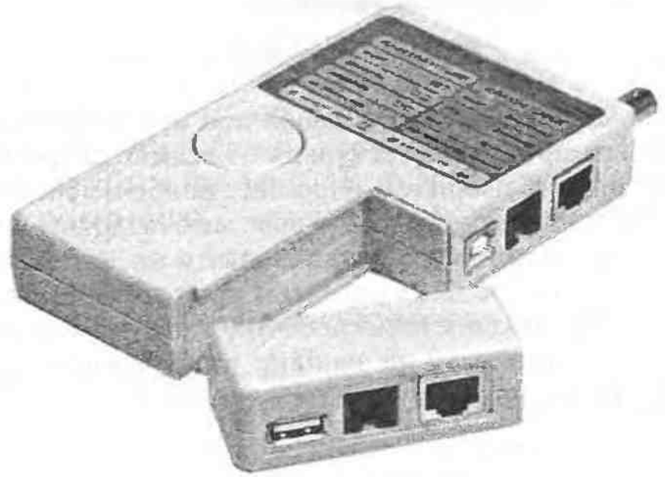


Рис. 2.31. Компактний тестер мережних ліній HL-NCTU

Тестер складається з активної і пасивної частин. Має вбудований BNC-термінал 25 / 50 Ом, індикацію прямого або кросового з'єднання, індикацію зарядки батареї живлення.

Мережні з'єднувальні засоби (розетки) мають відносно невелику надійність і в багатьох випадках виявляються "вузьким місцем" з'єднувального тракту в цілому. Найпоширеніший тип розетки для з'єднання складових комп'ютерних систем наведено на рис. 2.32.

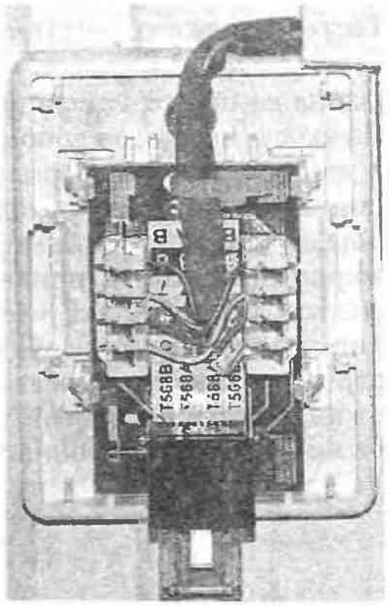


Рис. 2.32. Типова мережна розетка

У разі виявлення дефекту з'єднання або суттєвих відхилень параметрів лінії, яка проходила тестування, насамперед рекомендується заново здійснити підімкнення окремих проводів кабелю до затисків розетки. З огляду на те, що затиски мають досить специфічну конструкцію, цю операцію треба здійснювати за допомогою спеціального затискача. На рис. 2.33 наведено простий затискач проводів фірми *KRONE*. При використанні робочий інструмент головки затискача (рис. 2.34) забезпечує як надійне електричне з'єднання, так механічне закріплення проводу в затиску розетки.

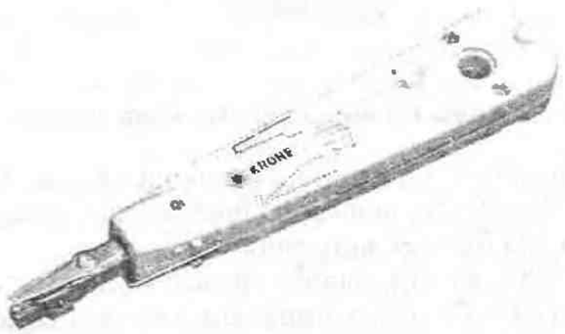


Рис. 2.33. Затискач проводів мережних ліній фірми *KRONE*

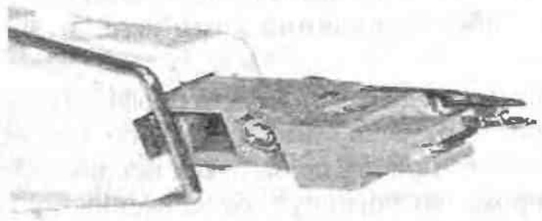


Рис. 2.34. Робочий інструмент головки затискача

2.4.4. Активне профілактичне обслуговування

Частота проведення активного профілактичного обслуговування комп'ютера залежить від стану навколишнього середовища і якості компонентів системи. Якщо комп'ютер установлено, наприклад, у заводському цеху або на автозаправній станції, то, можливо, доведеться проводити його чищення раз на три місяці, а то й частіше. Чищення комп'ютерів, що працюють в офісі, звичайно необхідно здійснювати раз на два роки. Однак якщо після року експлуатації, розкривши комп'ютер, можна знайти там шар пилу, значить, час між профілактичними роботами варто скоротити.

Далі розглядається ще одна операція, виконувана при профілактичному обслуговуванні, — періодичне резервне копіювання жорстких дисків.

2.4.4.1. Резервне копіювання системи

Один з основних етапів профілактичного обслуговування — резервне копіювання системи. Завдяки цій операції відновлюється працездатність системи в разі фатального апаратного збою. Для резервного копіювання необхідно придбати пристрій збереження високої ємності.

Природно, дискети для цього не підійдуть: вартість копіювання 4 Гбайт інформації буде просто «захмарною», крім того, виконання цієї операції вимагає значних затрат часу. Гідною альтернативою можуть служити пристрої на магнітній стрічці. Останнім часом вартість таких пристроїв постійно знижується, а ємність збільшується. Вибір пристрою для копіювання залежить від фінансових можливостей. Треба пам'ятати і про накопичувачі CD-RW, CD-R, Zip і Jazz, що також можна використовувати як пристрої резервного копіювання. Не має значення, як виконувати резервне копіювання системи, — головне, це зробити!

2.4.4.2. Засоби активного профілактичного обслуговування комп'ютера

Один із найбільш важливих елементів профілактичного обслуговування — регулярні і ретельні чищення. Пил, що осідає усередині комп'ютера, може стати причиною багатьох неприємностей. По-перше, він є теплоізолятором, що погіршує охолодження системи. Унаслідок цього скорочується термін служби компонентів і збільшується перепад температур при прогріві комп'ютера. По-друге, у пилу обов'язково наявні струмопровідні частки, що може призвести до виникнення витоків і навіть коротких замикань між електричними ланцюгами. І нарешті, деякі речовини, що містяться в пилу, можуть прискорити процес окислювання контактів, що спричинить порушення електричних з'єднань. У будь-якому разі чищення комп'ютера піде йому тільки на користь.

В усіх комп'ютерах, що не належать до типу ATX, і сумісних з ними моделях використовується примусове повітряне охолодження. Вентилятор, встановлений усередині, зовні або поруч із блоком живлення, витягує повітря з корпуса комп'ютера. При цьому тиск усередині корпуса виявляється нижче, ніж поза ним, і в комп'ютер крізь отвори в корпусі та шасі проникає повітря. Повітряні фільтри в таких випадках звичайно не встановлюють, оскільки важко забезпечити подачу повітря усередину корпуса через один вхідний отвір, який можна було б закрити фільтром.

У комп'ютерах ATX (а також NLX), часто застосовуваних у виробничих умовах, використовується інший принцип: вентилятор нагнітає повітря усередину корпуса, після чого той виходить назовні крізь отвори та щілини. Головна перевага цього методу полягає в тому, що єдиним каналом надходження повітря у внутрішній простір комп'ютера є вхідний отвір вентилятора. Тому очистити повітря досить просто: досить лише установити фільтр у горловині вентилятора. Фільтр, природно, доведеться періодично прочищати або замінять. Оскільки усередині корпуса тиск вище, ніж зовні, пил усередину потрапити не може, незважаючи на негерметичність. Усе повітря, що надходить у комп'ютер, проходить через вентилятор і крізь фільтр, що затримує частки пилу.

У більшості комп'ютерів, з якими вам доведеться мати справу, охолодження здійснюється за рахунок зниження тиску в їхніх корпусах. Установити які-небудь фільтри в ці комп'ютери неможливо, тому що повітря надходить у корпус крізь численні отвори. Природно, пил і різні хімічні речовини з навколишнього середовища потрапляють усередину й осідають там. Згодом такі «відкладення» можуть призвести до небажаних наслідків.

Дисководи найбільше зазнають забруднень, кожний з них виявляється великою «трубою», через яку постійно протікає повітря. Тому в них швидко накопичується величезна кількість пилу і небажаних хімічних сполук. З твердими дисками проблем менше. Вони мають герметичну конструкцію з одним клапаном, у якому встановлено повітряний фільтр. Чищення жорсткого диску зводиться до простого здування пилу з зовнішньої поверхні корпуса (усередині нічого протирати не треба).

Інструменти для розбирання і чищення комп'ютера

Для того щоб як слід почистити комп'ютер і усі встановлені в ньому плати, необхідні спеціальні інструменти і матеріали:

- розчин для чищення контактів;
- балончик зі стисненим повітрям;
- маленька щітка;
- поролонові тампони для чищення;
- заземлений наручний браслет.

Також можуть придатися:

- клейка стрічка;
- хімічно інертний герметик;
- силіконове мастило;
- малогабаритний пилосос.

Цих інструментів і хімікатів звичайно досить для виконання більшості профілактичних операцій.

Пристрої для видалення пилу

Істотною підмогою при «наведенні порядку» у системі може стати балончик (або компресор) зі стиснутим газом. З його допомогою пил і бруд можна просто здути з поверхні деталей. Раніше ці балончики заповнювалися фреоном, зараз — вуглеводнями або вуглекислим газом, що не завдає шкоди озоновому шару. Треба бути обережними: у процесі розширення газів при виході їх із сопла балона на останньому може накопичуватися великий електростатичний заряд. Справа в тому, що подібні пристрої використовуються для чищення кіно- та фотоапаратури, і вони не завжди відповідають вимогам електростатичної безпеки.

До пристроїв, у яких використовується стиснутий газ, належать балончики з охолодними рідинами. Вони призначені не для профілактики, а, скоріше, для ремонту. Справа в тому, що часто несправність компонента виявляється лише після його нагрівання, а охолодження на час відновлює його працездатність. Охолодною рідиною його можна швидко остудити. Якщо схема після цього починає працювати правильно, вважайте, що несправний елемент знайдено.

Пилососи

Іноді при «очисних роботах» перевага віддається пилососам. Зі стиснутим газом простіше працювати на маленьких ділянках. Пилососом можна «розгребти завали» у комп'ютері, покритому шарами пилу і бруду. Крім того, при застосуванні балончика пил, що ви здуватимете з одного компонента, тут же осідає на іншому, чого не трапляється за використання пилососа. При виїзному обслуговуванні у валізу з інструментами простіше покласти балончик зі стиснутим газом, а не пилосос, нехай навіть і маленький.

Існують пилососи, створені спеціально для обслуговування електронних пристроїв. Вони сконструйовані так, аби мінімізувати виникаючий електростатичний розряд. При використанні звичайного пилососа, в якому не передбачено захист від електростатичного розряду, необхідно вжити запобіжних заходів, наприклад надягти заземлений наручний браслет. Якщо в шланга пилососа — металева насадка, то слід бути обережним і не торкатися нею монтажних плат і компонентів.

Щітки і тампони

Перш ніж видаляти пил струмом стиснутого газу або пилососом, можна зняти її невеликою щіточкою (цілком підійдуть косметичні, а також щіточки для ретуші фотографій або пензлі для малювання).

Чистити щітками найкраще корпуси блоків, лопаті вентиляторів, грати отворів забору повітря і клавіатуру.

Контакти рознімань, голівки дисководів і інші важливі вузли звичайно протирають тампонами з матеріалів на кшталт поролону або штучної замші, що не залишають після себе волосків і пилу. Такі тампони набагато дорожчі за ватяні. Але останніми, за всієї їхньої дешевизни, усе-таки краще не користуватися, оскільки буквально на усьому, з чим вони стикаються, залишаються волокна бавовни, що за певних умов можуть стати провідниками або прилипнути до голівок дисководів і подряпати поверхню гнучкого диска. Тампони для чищення з поролону або замші можна придбати в більшості магазинів, що торгують апаратурою і радіодеталями.

Не слід терти контакти ластиком. Багато хто рекомендує очищувати бруд і оксидні плівки з друкованих контактів м'яким олівцевим ластиком. Як показали експерименти, цей спосіб не підходить з кількох причин. По-перше, при терті ластику об контакти утворяться електростатичні заряди. Вони можуть вивести з ладу мікросхеми, установлені на платах. Чистити контакти плат краще «вологим» способом (використовуючи відповідні рідини).

По-друге, навіть при використанні найм'якших ластиків захисне золоте покриття частково стирається, відкриваючи повітрю і волозі доступ до основного матеріалу контактів. Деякі фірми випускають спеціальні тампони, заздалегідь просочені відповідною рідиною, зі змащувальними добавками. Вони безпечні як з погляду електростатичних розрядів, так і з погляду збереженості золотого покриття контактів.

Хімічні засоби чищення

Для чищення комп'ютерів і інших електронних пристроїв використовуються хімічні речовини. Їх можна розділити на такі основні групи:

— універсальні очисники;
— засоби для чищення і змащення контактів.

Силіконове мастило

Це мастило застосовують замість машинних олій для чищення механізмів фіксації дискет у накопичувачах, напрямних, якими переміщаються блоки голівок дисководів, або напрямної друкувальної голівки принтера.

Перевага силікону полягає в тому, що він згодом не застигає і до нього не прилипає пил. Кількість нанесеного мастила має бути мінімальною, краплі і патьоки зовсім неприпустимі. Поява мастила в непередбачених для цього місцях (наприклад, на голівках накопичувачів) може призвести до найнеприємніших наслідків. Для точкового нанесення мастила краще користуватися пластмасовою зубочисткою, а якщо треба змастити поверхню (наприклад, напрямну голівку принтера) — губчатим тампоном.

Треба мати на увазі, що при виконанні деяких операцій можуть утворюватися статичні заряди. Тому обов'язково заземлюйте в цих випадках усе, що тільки можна (у тому числі і себе), щоб не вивести з ладу мікросхеми на платах.

Розбирання і чищення

Для того щоб як слід почистити комп'ютер, його необхідно хоча б частково розібрати. Деякі особливо старанні шанувальники чистоти доходять до того, що знімають системну плату. Звичайно, при цьому ви одержите прекрасний доступ до інших вузлів, але, на мій погляд, досить довести розбирання до тієї стадії, коли системна плата виявиться цілком відкритою.

Вам доведеться вийняти всі знімні плати адаптерів і дисководи. Хоча голівки дисководів можна протерти за допомогою чистячої дискети не знімаючи кришку комп'ютера, можливо, вам захочеться

зробити більш ґрунтовне «збирання». Крім голівок, можна протерти і змастити механізм фіксації дискети, а також почистити плати керування і рознімання. Для цього дисковод звичайно доводиться витягати з комп'ютера.

Ті самі операції виконують і з жорстким диском: чистять плати і рознімання, а також змащують заземлювальну пластинку. Для цього накопичувач на жорсткому диску доведеться вийняти. Про всяк випадок, перш ніж робити це, створіть резервну копію даних, що зберігаються на диску.

Установлення мікросхем на свої місця

Під час профілактичного обслуговування дуже важливо усунути наслідки термічних зсувів мікросхем. Оскільки комп'ютер при вмиканні і вимиканні нагрівається й остигає (отже, його компоненти розширюються і стискуються), мікросхеми поступово «виповзають» з гнізд. Тому доведеться знайти усі компоненти, з якими таке сталося, і поставити їх на місця.

У більшості комп'ютерів мікросхеми пам'яті містяться в гніздах або входять до складу модулів SIMM або DIMM. Ці модулі фіксуються в розніманнях за допомогою спеціальних засувок. У модулів SIPP (аналогічних SIMM, але зі штирьовими, а не друкованими виводами) таких засувок немає, тому вони іноді «вилазять» зі своїх гнізд. Але першими кандидатами на «виповзання» є звичайні мікросхеми пам'яті, встановлювані в гнізда. Крім зазначених інтегральних схем, у гніздах можуть бути розміщені мікросхеми ROM, мікропроцесор і співпроцесор. Всі інші інтегральні схеми в більшості комп'ютерів установлюються паянням.

Утім, можливі варіанти. Компоненти, що в одному комп'ютері встановлені в гнізда, в іншому можуть бути просто впаяні (навіть якщо ці комп'ютери виготовлені однією і тією самою фірмою). Подібні розходження звичайно пов'язані з такою прозаїчною обставиною, як наявність на заводі певних мікросхем. Якщо до моменту збирання плати їх на складі не виявилося, щоб не зупиняти виробництво, замість них установлюються порожні гнізда. Коли необхідні мікросхеми з'являються, їх просто швидко ставлять у гнізда — і плати готові. У багатьох нових комп'ютерах мікропроцесори встановлюються в гнізда ZIF (*Zero Insertion Force* — з нульовим зусиллям вставляння) з важільцем, за допомогою якого можна затиснути або звільнити одразу усі виводи встановленої мікросхеми. Як правило, з гнізд типу ZIF мікросхеми не «виповзають».

Для того щоб поставити мікросхему в гніздо, треба натиснути на неї зверху великим пальцем, обов'язково притримуючи при цьому

плату долонею зі зворотного боку. З великими мікросхемами треба поводитися більш обережно. Їх установлюють, по черзі натискаючи спочатку з одного, а потім з іншого боку, поки вони повністю стануть на місце (так звичайно поводяться з процесором і співпроцесором). При переміщенні мікросхеми вниз часто виразно чується скрип. Оскільки при цьому до плат докладаються значні зусилля, їх краще виймати з рознімань або з корпуса.

Вищесказане насамперед стосується системних плат. У жодному разі не можна надавлювати на мікросхеми, якщо немає можливості притримати плату іншою рукою зі зворотного боку, бо вона прогнеться, а при занадто великому зусиллі може зламатися, перш ніж мікросхема стане на місце. Пластмасові стійки, на які встановлюється системна плата, рознесені надто далеко і не можуть запобігти її прогинанню за великого натискання. Тому, перш ніж поправляти мікросхеми на системній платі, вийміть її з корпуса — інакше не буде можливості підтримувати її знизу.

Не дивуйтесь, якщо приблизно через рік після того, як ви встановите мікросхеми на місце, вам доведеться робити це знову. Це цілком нормальне явище.

Чищення плат

Для чищення плат і рознімань вам знадобляться описані вище тампони й розчини.

Спочатку треба очистити плати від пилу і бруду, а потім займатися встановленими на них розніманнями. Плати найкраще чистити за допомогою спеціального пилососа або балончика зі стиснутим газом. Останній особливо ефективний для здування пилу з плат, на яких установлена велика кількість компонентів.

Також дуже важливо видути пил із блоку живлення, при цьому звернути особливу увагу на отвори, крізь які вентилятор протягує повітря. Для цього не потрібно розбирати блок живлення, досить лише продути його, направивши струмінь стисненого повітря у вихідний отвір вентилятора. Таким чином пил з внутрішніх компонентів блоку живлення, лопаті вентилятора і грат, що їх закривають, буде видалено.

Чищення контактів рознімань

Протирати контакти рознімань потрібно для того, щоб з'єднання між вузлами і компонентами системи були надійними. Варто звернути увагу на рознімання розширення, електроживлення, підімкнення клавіатури і динаміка, розташовані на системній платі. Що стосується плат адаптерів, то на них треба протерти друковані рознімання,

що вставляються в слоти на системній платі, і всі інші рознімання (наприклад, рознімання, установлені на зовнішній панелі адаптера).

Чищення проводиться таким чином. Необхідно змочити тампон розчином для чищення. Якщо користуватися аерозолем, то треба нанести на тампон таку кількість рідини, щоб вона почала з нього капати.

Розчин не треба заощаджувати, частіше змочуйте тампон і протирайте рознімання як слід. Не переймайтеся тим, що краплі рідини залишаються на поверхні системної плати. Ці розчини безпечні як для самої плати, так і для встановлених на ній компонентів.

Починайте чищення з позолочених контактів рознімань, а потім переходьте до всього іншого. Протріть рознімання для підімкнення клавіатури, динаміка, живлення, батареї, а також ділянки поверхні, з якими контактують голівки гвинтів, що кріплять і з'єднують загальну шину системної плати з шасі електрично.

На платах адаптерів особливо ретельно необхідно протерти контакти друкованих рознімань, що вставляються в рознімання на системній платі. До їхніх позолочених контактів, як правило, доторкаються, коли беруть у руки плату адаптера. При цьому контакти покриваються жирними плямами, що погіршує контакт адаптера з системною платою. Для протирання саме таких рознімань непогано було б використовувати засіб для чищення з додаванням струмопровідного мастила, що, по-перше, сприяло б до зниженню необхідного зусилля при установленні плати адаптера в слот, а по-друге, захистило би контакти від окислювання.

Тим самим розчином можна протерти рознімання плоских кабелів і решту з'єднувачів у комп'ютері, насамперед рознімань інтерфейсних кабелів накопичувачів на гнучких і жорстких дисках, друкованих платах керування дисководів, а також рознімань живлення.

Чищення клавіатури та миші

Клавіатура та миша завжди сильно забруднюються. Тому клавіатуру треба періодично чистити пилососом. Можна також перевернути клавіатуру клавішами вниз і продути її струменем стисненого повітря. Це допоможе позбутися більшої частини накопиченою бруду, а разом із тим і від неприємностей, пов'язаних з поганими контактами в клавішних перемикачах.

Якщо яка-небудь клавіша, незважаючи на це, «залипне» або контакт із нею стане ненадійним, треба капнути до її контактного вузла невелику кількість очисника. Проблеми з поганими контактами і «залипанням» клавіш не виникають, якщо періодично чистити клавіатуру за допомогою пилососа або балончика зі стисненим повітрям.

У більшості випадків для того, щоб почистити мишу, досить відвернути фігурну шайбу (кришку), що закриває отвір з кулькою, і витрусити її з гнізда. Протерти кульку якою-небудь очисною рідиною. Використовувати для цього очисник з мастилом не можна, бо кулька буде ковзати, а не котитися по столу. Після цього прочистити щіточкою або тампоном, змоченим в очиснику, ролики, з якими контактує кулька всередині корпуса миші.

Існує пристрій позиціонування, що вимагає мінімальної уваги, — це *Track point*, створений IBM, і подібні йому пристрої, представлені іншими виробниками, наприклад *Glide point* фірми *Alps*. Ці пристрої цілком герметичні і керують покажчиком за допомогою спеціальних датчиків. Чищення зводиться до простого протирання поверхні ганчіркою з використанням слабкого очисного розчину.

2.4.4.3. Профілактичне обслуговування жорстких дисків

Щоб гарантувати збереженість даних і підвищити ефективність роботи жорсткого диска, необхідно час від часу виконувати деякі процедури з його обслуговування. Існує також кілька простих програм, за допомогою яких можна якоюсь мірою застрахувати себе від утрати даних. Ці програми створюють резервні копії тих критичних зон жорсткого диска (і в разі необхідності відновлюють їх), при ушкодженні яких доступ до файлів стає неможливим.

Дефрагментація дисків

У міру того як ви записуєте файли на жорсткий диск і видаляєте їх, багато з них фрагментуються, тобто розбиваються на безліч розкиданих по всьому диску частин. Періодична дефрагментація файлів вирішує одразу дві задачі. По-перше, якщо файли займають безперервні області на диску, то переміщення голівок при їх зчитуванні і записуванні стає мінімальним, що зменшує знос привода голівок і самого диска. Крім того, істотно збільшується швидкість зчитування файлів з диска. По-друге, при серйозних ушкодженнях таблиць розміщення файлів (*File Allocation Table* — FAT) і кореневого каталогу дані на диску легше відновити, якщо файли записані як одне ціле. Якщо ж вони розбиті на безліч фрагментів, то, не звертаючися до FAT і структури каталогів, практично неможливо визначити, якому файлу належить той або інший фрагмент. В інтересах збереження інформації треба виконувати дефрагментацію жорсткого диска раз на тиждень або після кожної операції резервного копіювання.

У більшості програм дефрагментації передбачені такі функції:

— дефрагментація файлів;

— ущільнення файлів (упорядкування вільного простору);
— сортування файлів.

Основною операцією є дефрагментація, але в більшості програм передбачене й ущільнення файлів. Дефрагментація не виконується автоматично, а повинна бути зазначена особливо, оскільки на неї затрачується додатковий час. При її проведенні усі файли, записані на диску, переміщаються до його початку, а вільний простір розташовується наприкінці. Завдяки цьому записувані згодом файли не фрагментуються і весь вільний простір являє собою єдину область, достатню для запису будь-якого файла без його розбивки на частини.

Остання операція — сортування файлів — не є життєво необхідною, але передбачена в багатьох програмах дефрагментації. Виконується вона дуже довго, але на швидкість доступу до даних практично не впливає. Безумовно, сортування має деякий сенс, оскільки у процесі відновлювання даних стає відомим, у якому порядку розташовувалися файли до моменту аварії. Хоча знати це і не обов'язково — цілком достатньо того, щоб усі файли були дефрагментовані. Порядок їх розташування в цьому випадку не має значення. Сортування файлів передбачене не в усіх програмах дефрагментації, оскільки затрачуваний час не відповідає результатові.

Для різних операційних систем існують різні програми дефрагментації. До складу Windows, починаючи з Windows *9x* і вищих модифікацій, входить програма, що працює з файловими системами FAT 16 і FAT 32. Вона являє собою графічний додаток, що може виконуватись у фоновому режимі. Тому дана програма дефрагментації краща за інші. Під час її роботи можна викликати вікно з докладною інформацією про процес дефрагментації або обмежитися мінімальною інформацією про етапи процесу.

Слід пам'ятати, що програми дефрагментації для файлових систем FAT 16 і FAT 32 несумісні. Тому не можна запускати програми *Scan Disk for* DOS або *Norton Disk Doctor* у середовищі Windows *XP* — наслідки можуть бути непередбачені!

2.4.5. Пасивне профілактичне обслуговування

Під пасивною профілактикою мають на увазі створення прийнятних для роботи комп'ютера загальних зовнішніх умов. Треба враховувати фізичні впливи: температуру навколишнього повітря, тепловий удар при вмиканні і вимиканні системи, пил, дим, а також такі немаловажні чинники, як вібрація й удари. Крім того, дуже важливі електричні впливи: електростатичні розряди, перешкоди в ланцюгах живлення та радіочастотні перешкоди.

2.4.5.1. Робоче місце

Кінцева мета будь-якої профілактики — збереження устаткування (і вкладених у нього засобів). Комп'ютери цілком надійно працюють у сприятливих для людини умовах. Однак вони, як правило, швидко псуються в разі зневажливого ставлення до них.

До робочого місця, де планується встановити комп'ютер, висуваються певні вимоги, основними з яких є:

— мінімізація пилу в приміщенні, а в навколишньому повітрі — тютюнового диму;
— на робочому місці не повинно бути прямого сонячного світла;
— перепади температури мають бути якомога меншими.

Вмикати комп'ютер потрібно в надійно заземлені розетки, напруга в мережі повинна бути стабільною, без перепадів і перешкод.

2.4.5.2. Нагрівання й охолодження комп'ютера

Коливання температури несприятливо позначаються на стані комп'ютера. Тому, щоб комп'ютер працював надійно, температура в приміщенні повинна бути сталою.

При коливанні температури можуть потріскатися або відшаруватися струмопровідні площадки на друкованих платах, зруйнуватися паяні з'єднання. За підвищеної температури прискорюється окислювання контактів, можуть зіпсуватися мікросхеми й інші електронні компоненти.

Коливання температури можуть позначитися і на роботі жорстких дисків.

Для будь-яких електронних пристроїв, зокрема й для комп'ютерів, указується припустимий діапазон температур. Більшість фірм-виробників наводить ці дані в документації на виріб. У ньому мають бути зазначені два діапазони температур: при експлуатації і при зберіганні. Наприклад, для більшості комп'ютерів фірми IBM ці діапазони такі:

— при експлуатації: від + 15 до + 32 °С;
— при зберіганні: від + 10 до + 43 °С.

Задля збереження як самого диска, так і записаних на ньому даних оберігайте його від різких перепадів температури. Якщо ж такий перепад неминучий (наприклад, у разі перенесення комп'ютера узимку з морозу в тепле приміщення), перед його ввімкненням треба витримати деякий час для прогрівання його до кімнатної температури. Найчутливішою частиною комп'ютера є накопичувачі на магнітних дисках, оскільки в них може конденсуватися волога і при спробі ввімкнення накопичувач вийде з ладу.

2.4.5.3. Цикли вмикання та вимикання

Як зазначалося вище, коливання температури несприятливо впливають на компоненти комп'ютера. Існує два очевидні способи звести до мінімуму коливання температури в системі: або назавжди залишити комп'ютер увімкненим, або ніколи його не вмикати. Навряд чи другий варіант взагалі прийнятний. Тому, якщо головною і єдиною метою є продовження терміну служби системи, слід тримати комп'ютер постійно ввімкненим. Оскільки у реальному житті враховуються такі обставини, як, наприклад, вартість електроенергії, пожежна безпека і т.п., цикл увімкнення і вимкнення треба проводити якомога рідше.

Найчастіше в момент увімкнення виходять з ладу блоки живлення. Струмові перевантаження, що виникають під час увімкнення, пов'язані, наприклад, з розгоном двигунів, сила струму яких значно перевищує споживацькі джерела живлення в стаціонарному режимі. Протягом перших секунд роботи блок живлення віддає (і, отже, розсіює) велику потужність, особливо якщо одночасно розкручуються двигуни одразу кількох накопичувачів, для яких характерні найбільш високі значення пускових струмів. Часто це служить причиною перевантаження як вхідних, так і вихідних компонентів блоку живлення.

Певну загрозу для компонентів комп'ютера становлять електростатичні заряди. Найбільш небезпечні вони узимку, за низької вологості повітря, а також у районах із сухим кліматом. У цих умовах працюючи з комп'ютером необхідно вжити спеціальних запобіжних заходів.

Електростатичні явища поза корпусом системного блоку рідко призводять до серйозних наслідків, але на шасі, клавіатурі або просто поруч із комп'ютером сильний розряд може спричинити порушення при перевірці парності (у пам'яті) або «зависання» комп'ютера. Як правило, усі ці проблеми виникають тому, що кабель живлення комп'ютера погано заземлений. Для підімкнення системи до мережі потрібно користуватися вилкою з трьома штирями, а заземлення розетки має бути надійним.

Особливі запобіжні заходи необхідно вживати при відкритті системного блоку або роботі з окремими вузлами і платами, витягнутими з комп'ютера. Якщо вчасно не відвести статичний заряд, що набрався, можна вивести з ладу багато компонентів комп'ютера. Усякого разу, виймаючи з корпуса плати або адаптери, для вирівнювання електростатичного потенціалу беріться за ділянки, з'єднані з загальним проводом, наприклад за кронштейни.

Питання для самоперевірки

1. З чого складається необхідний набір інструментальних засобів для пошуку несправностей і ремонту КС?
2. Для чого використовуються вимірювальні прилади загального призначення — мультиметри та логічні пробники?
3. Які задачі вирішують тестери мережних розеток і кабелів?
4. У чому полягає принцип активного профілактичного обслуговування КС?
5. Які засоби активного профілактичного обслуговування комп'ютера рекомендовано для персональних комп'ютерів?
6. Що відноситься до пасивних методів профілактичного обслуговування КС?
7. З чого складається профілактичне обслуговування жорстких дисків у КС?

2.5. Охорона праці при експлуатації комп'ютерних систем та мереж

2.5.1. Загальні питання безпеки праці

Технічні засоби (ТЗ) КС та М є складними комплексами різноманітних технічних і програмних засобів. Технічні засоби містять у своєму складі радіоелектронне, електронне, електротехнічне, механічне й інше обладнання. Технічна експлуатація цього обладнання пов'язана з деякою небезпекою й шкідливістю, які слід враховувати.

Зупинимося на основних термінах і визначеннях.

Безпека виробничого процесу — це властивість під час його протікання зберігати безпечний стан у заданих параметрах протягом встановленого часу.

Безпека виробничого обладнання — це властивість обладнання зберігати безпечний стан при виконанні заданих функцій у визначених умовах упродовж встановленого часу.

Заходи безпеки — це система організаційних і технічних заходів, що мають на меті забезпечення безпеки та збереження здоров'я людей при експлуатації техніки.

Норми безпеки — гранично допустимі безпечні значення міцності і зносу відповідальних елементів, робочих і випробувальних навантажень, напруги, тиску, а також строку проведення

періодичних випробувань обладнання, захисних засобів і пристроїв захисту з метою визначення можливості їх подальшого використання.

Охорона праці — це система законодавчих актів, відповідних соціально-економічних, технічних, гігієнічних і організаційних заходів, які забезпечують безпеку, збереження здоров'я і працездатність людини в процесі роботи.

Техніка безпеки — система організаційних і технічних заходів та засобів, які запобігають впливу небезпечних виробничих факторів на працівників.

Правила безпеки — обов'язкові накази, які визначають безпечні способи роботи на техніці і правильне використання засобів захисту.

Виробнича санітарія — система організаційних, гігієнічних, санітарно-технічних заходів і засобів, які запобігають впливу шкідливих виробничих факторів на працівників.

Нещасний випадок — травма (захворювання чи отруєння), зумовлена раніш невідомим чи непередбаченим конструкторською або експлуатаційною документацією небезпечним фактором.

Виробнича травма — травма (порушення анатомічної цілості або фізіологічних функцій окремих органів, а іноді і всього організму), що виникла у працівника на виробництві внаслідок невиконання вимог техніки безпеки.

Виробничий фактор — фактор, вплив якого призводить до захворювання (шкідливий виробничий фактор) чи до травми (небезпечний виробничий фактор).

Система безпечної експлуатації — сукупність технічних засобів безпеки, встановлених правил виконання шкідливих робіт, організаційно-технічних, санітарно-гігієнічних заходів і органів, які відповідають за їх проведення, що забезпечує безаварійність і виключає можливість завдання здоров'ю короткострокової чи довгострокової шкоди в процесі роботи на техніці.

Засоби захисту — засоби, які використовуються для запобігання чи зменшення впливу шкідливих небезпечних виробничих факторів на працівників.

При експлуатації КС та М необхідно керуватися державними стандартами і встановленими правилами.

В обов'язки керівників усіх ступенів входить забезпечення безаварійної експлуатації техніки і норм безпеки, усунення умов, які призводять до нещасних випадків і травм. Особовий склад, пов'язаний з експлуатацією КС та М, зобов'язаний знати і безумовно виконувати основні положення безпечної роботи.

2.5.2. Небезпечні фактори на об'єктах КС та М

Небезпечні фактори на об'єктах КС та М пов'язані з використанням електричного струму в широкому діапазоні діючої напруги, засобів, що випромінюють електричні й електромагнітні поля, а також можливістю виникнення пожежної ситуації.

Для електроживлення ТЗ КС та М використовуються мережі змінного струму частоти 50 Гц з напругою 220 і 380 В. Це мережі первинного електроживлення. Від них живиться насамперед все технологічне обладнання КС та М, а також допоміжне кондиціонерне і вентиляційне обладнання, антенні пристрої. В технологічному обладнанні використовується велика кількість джерел вторинного електроживлення, які забезпечують постійний струм різної напруги.

У процесі роботи можуть виникнути умови, коли під вплив електричного струму може потрапити представник технічного персоналу чи користувач.

При протіканні через організм людини електричний струм справляє термічний, електролітичний, механічний і біологічний вплив. Термічний вплив проявляється в опіках ділянок тіла, в прогріві до високої температури кровоносних судин, нервів, серця, мозку й інших органів, які знаходяться на шляху протікання струму, що призводить до серйозних функціональних порушень.

Електролітична дія струму проявляється в розкладі органічної рідини, зокрема крові, біологічна — в подразненні та збудженні живих тканин організму, що супроводжується судорожними скороченнями м'язів, зокрема м'язів серця і легенів; механічна — в розшаруванні, розриві й інших пошкодженнях тканин організму, кровоносних судин.

Людина починає відчувати протікання змінного струму через організм частотою 50 Гц силою 0,5—1,5 мА і постійного струму силою 5—7 мА. Важкопереносимою є сила струму 10—15 мА. Струм силою 10—15 мА частоти 50 Гц і постійного струму 50—80 мА називається пороговим невідпускаючим струмом.

При силі струму 25—50 мА частотою 50 Гц дуже утруднюється дихання людини, струм більше 50 мА порушує роботу легень і серця.

У табл. 2.4 наведено найбільші значення допустимих для людини струмів частотою 50 Гц залежно від тривалості часу їх протікання.

Таблиця 2.4

ДОПУСТИМІ ЗНАЧЕННЯ СТРУМІВ КРІЗЬ ТІЛО ЛЮДИНИ

Тривалість протікання струму, с	0,2	0,5	0,7	1	3—30	Більше 30
Сила струму, мА	250	100	75	65	6	1
Опір тіла, Ом	700	1000	1065	1150	3000	6000
Напруга, В	175	100	80	75	18	6

Наведені в табл. 2.4 значення сили струму не є безпечними й приймаються як допустимі з достатньо малою ймовірністю ураження.

Небезпека ураження струмом росте з ростом сили струму, що протікає. Збільшення частоти в межах 0—50 Гц збільшує небезпеку ураження, але надалі її підвищення супроводжується зниженням небезпеки ураження, що повністю зникає при частоті 450—500 кГц.

На цих частотах струм не може викликати смертельного ураження внаслідок припинення роботи серця і легенів. Однак цей струм зберігає небезпеку опіків як при виникненні електричної дуги, так і при протіканні через тіло людини.

Постійний струм у 4—5 разів безпечніше змінного струму частотою 50 Гц. Чим більше опір тіла людини, тим менше вона підпадає небезпеці. При сухій, чистій і непошкодженій шкірі за напруги 15—20 В опір тіла становить 3—100 кОм. При пошкодженій шкірі опір падає до 1—10 кОм, а в разі відсутності всього зовнішнього шару шкіри — до 500—700 ОМ і менше.

Люди з пошкодженням шкіри, хворобою серцево-судинної системи, органів внутрішньої секреції, легенів, нервовими захворюваннями мають підвищену вразливість електричним струмом.

Джерела електричного струму з напругою 48 В і більше вважаються небезпечними в нормальних умовах використання. В приміщеннях з підвищеною вологістю, за наявності хімічних речовин небезпечними вважаються джерела з напругою більше 12 В.

На підприємствах використовуються спеціальні заходи захисту від можливого потрапляння людини під вплив електричного струму. До таких заходів належать: захисне заземлення, захисні засоби, блискавкозахист споруд та ін.

Захисне заземлення здійснюють для забезпечення безпеки людей при замиканні струмопровідних частин обладнання на корпус. Для забезпечення нормальних режимів роботи обладнання використову-

ється робоче заземлення. Для захисту обладнання від перенапруги і блискавки використовують блискавкозахисне заземлення.

Заземленням називається навмисне контактне з'єднання металічних частин конструкції з заземлювальним пристроєм. Він складається з заземлення і заземлювальних провідників. Заземленням називається провідник чи група електрично-з'єднаних провідників, які мають безпосередній контакт із землею. Заземлювачі бувають природні і штучні. Природними є прокладені в землі металеві конструкції, не призначені для заземлення, але які можуть виконувати таку роль (трубопроводи, арматура залізобетонних споруд, окрім трубопроводів для горючих і вибухових речовин і газів, а також трубопроводів, покритих ізоляцією для захисту від корозії).

Основною електротехнічною характеристикою заземлювального пристрою є опір розтіканню струму, який визначається як відношення напруги на пристрої до струму, який стікає в землю.

При проходженні струму з заземлювача в землю на її поверхні виникає електричний потенціал, величина якого зменшується в міру віддалення від заземлювача. Людина, що перебуває біля заземлення, може потрапити під дію електричного потенціалу. Різниця потенціалів двох точок поверхні землі, яких одночасно може доторкнутися людина, називається напругою на тілі при ході (крокова напруга). Чим ширше крок, тим більша крокова напруга, тим більше небезпека ураження струмом.

Для захисту технічного персоналу від ураження електричним струмом використовуються захисні засоби. Вони поділяються на ізолювальні, загороджувальні і допоміжні.

Ізолювальні захисні засоби забезпечують електричну ізоляцію технічного персоналу від струмопровідних чи заземлювальних частин обладнання і від землі. До них належать: інструмент з ізольованими рукоятками, ізолювальні штанги, ізолювальні підставки, ізолювальні покриття та ін. Всі ізолювальні захисні засоби періодично перевіряються на ізоляційні властивості.

На практиці широко використовуються автомати захисту електромережі й інші запобіжники. В разі виникнення перенавантаження в мережі збільшується струм, і автомат захисту відключає мережу від навантаження.

Для прийому електричного розряду блискавки і відводу його струму в землю використовуються блискавковідводи. Вони складаються з опори, блискавкоприймача, струмовідводу і заземлювача. Найчастіше використовуються стержневі і тросові блискавковідводи. Сила струму в блискавковідводі може досягати 100—200 тис. А.

У приміщеннях, де розташоване обладнання КС та М, можливе загорання тканин, пластмасових виробів, виробів з оргскла, дерева, паперу, синтетичних і гумових виробів, ізоляції провідників, кабелів, пально-мастильних, спиртових і лакофарбових матеріалів. Причин для цього може бути багато, якщо не виконувати правила пожежної безпеки. Таке загорання може виникнути в результаті неохайності обслуговуючого персоналу при користуванні паяльниками, при курінні в недозволених місцях, унаслідок несправності електромережі, перегріву елементів конструкції апаратури, силових кабелів, виникнення електричної дуги в разі короткого замикання, удару блискавки тощо.

На кожному об'єкті обслуговуючий персонал повинен знати і строго виконувати правила пожежної безпеки. Але мають бути і необхідні засоби пожежегасіння. На випадок пожежі повинні бути передбачені заходи оперативного вимикання електромережі, запасні виходи для рятування людей і винесення необхідного обладнання.

Відомі випадки, коли виникали пожежі в приміщеннях, де було встановлене радіоелектронне обладнання і двигуни внутрішнього згорання. Були випадки виникнення пожежі навіть в робочих приміщеннях митних установ.

Виникнення таких пожеж може вивести з ладу на тривалий час митну установу чи її обладнання, що може призвести до значних економічних збитків.

Для пожежегасіння повинні бути спеціальні вогнегасники. Використання води для гасіння пожежі в приміщеннях, де встановлено електро- і радіоелектронне обладнання, недопустиме. Для цього не можна користуватись і кислотно-лужними вогнегасниками.

В електроустаткуванні, яке знаходиться під напругою до 380 В при температурі повітря від – 25 °С до + 50 °С, рекомендується використовувати вуглекислотні вогнегасники типів ОУ-5, ОУ-8, ОУ-25, ОУ-80, вуглекислотні бром-етилові вогнегасники типів ОУБ-3 і ОУБ-7.

Ці вогнегасники мають постійно бути в легкодоступному місці приміщення і перевірятися не менше одного разу на рік.

2.5.3. Шкідливі фактори на об'єктах КС та М

Радіотехнічні й електронні засоби КС та М випромінюють електромагнітні хвилі, які негативно впливають на навколишнє середовище і певною мірою шкідливі для людини. На цих засобах працюють люди і для них мають бути створені умови, що найменшою мірою впливають на здоров'я.

Радіохвилі за діапазоном класифікуються згідно з табл. 2.5.

Таблиця 2.5

КЛАСИФІКАЦІЯ РАДІОЧАСТОТ І РАДІОХВИЛЬ

Міжнародна класифікація			Вітчизняна класифікація, що використовується в практиці гігієнічного нормування	
Діапазон	Радіочастоти	Радіохвилі	Радіочастоти	Радіохвилі
5-й	Низькі — 30—300 кГц	Кілометрові — 10—1 км	Високі — 100 кГц—30 МГц	Довгі — 10—1 км
6-й	Середні — 0,3—3 МГц	Гектометрові — 1—0,1 км		Середні — 1—0,1 км
7-й	Високі — 3—30 МГц	Декаметрові — 100—10 м		Короткі — 100—10 м
8-й	Дуже високі — 30—300 МГц	Метрові — 10—1 м	Ультрависокі — 30—300 МГц	Ультракороткі — 10—1 м
9-й	Ультрависокі — 0,3—3 Ггц	Дециметрові — 1—0,1 м		Дециметрові — 1—0,1 м
10-й	Надвисокі — 3—30 ГГц	Сантиметрові — 10—1 см	Надвисокі — >300 МГц	Сантиметрові — 10—1 см
11-й	Вкрай високі — 30—300 ГГц	Міліметрові — 10—1 мм		Міліметрові — 10—1 мм

Джерела радіохвиль за потужністю випромінювання поділяють на чотири групи. Радіостанції довгих, середніх і коротких хвиль мають малу (менше 5 кВт), середню (від 5 до 25 кВт), велику (від 25 до 100 кВт) і надвелику потужність (більше 100 кВт).

Максимально допустимий рівень електромагнітного опромінювання в населених пунктах не повинен перевищувати для хвиль:

- довгих — 20 В/м; середніх — 10 В/м; коротких — 4 В/м; ультракоротких — 2 В/м;
- сантиметрових і міліметрових за цілодобового опромінювання — 5 мкВт/см^2 (щільність потоку енергії).

Контроль рівнів електромагнітного поля здійснюють органи санітарно-епідеміологічної служби Міністерства охорони здоров'я на стадіях проектування, впровадження і експлуатації.

Вимірювання рівня енергії електромагнітного поля виконується за допомогою приладів, наведених в табл. 2.6.

Таблиця 2.6

ПРИЛАДИ ДЛЯ ВИМІРЮВАННЯ ЕЛЕКТРОМАГНІТНОГО ПОЛЯ

Тип приладу	Робочий діапазон частот	Межі вимірювання
ИЭМП-1	100 кГц—30 МГц	4—1500 В/м
ПО-1	0,15—16,7 ГГц	0,016 мкВт/см^2—30 мВт/см^2
ПЗ-2	200 кГц—300 кГц	0,5—3000 В/м
ПЗ-9	0,30—37,5 ГГц	0,016 мкВт/см^2—30 мВт/см^2
ПЗ-13	150 МГц—16,7 ГГц	0,5—10000 мкВт/см^2
П4-5А	20—150 МГц	0,001—100 В/м
П4-12А	0,15—30 МГц	1*(102—105) мкВ/м
П4-13А	30—300 МГц	1*(10—105) мкВ/м

Вимірювання напруженості електромагнітного поля виконують на відстанях 50, 100, 300, 500, 1000, 2000, 3000, 5000 м від антени.

Місця для вимірювання вибирають залежно від діаграми спрямованості антени в горизонтальній площині в напрямі максимуму опромінювання головної пелюстки діаграми направленості, а також бокових і задніх пелюсток.

Під впливом електромагнітного опромінювання в організмі людини порушується процес терморегуляції, якщо це опромінювання перевищує допустимі норми. Найбільш чутливі до опромінювання мозок, очі, нирки, шлунок та інші органи людини. Під впливом електромагнітних хвиль порушується хімічний склад тканин, розриваються міжмолекулярні зв'язки. Довгостроковий систематичний вплив електромагнітних хвиль може призвести до функціональних змін в організмі, передусім в нервовій системі. Виникає головний біль, порушується сон, підвищується втомлюваність, роздратованість. В організмі накопичуються й функціональні зміни, викликані біологічною дією електромагнітних хвиль, але вони є зворотними.

У діапазонах міліметрових, сантиметрових і дециметрових хвиль гранично допустима щільність опромінювання залежить від тривалості робочого дня і дорівнює:

– до 5 мкВт/см^2 протягом 24 год; до 10 мкВт/см^2 протягом 6 год; — до 100 мкВт/см^2 протягом 2 год.;

– до 1000 мкВт/см^2 протягом 15 хв робочого дня при обов'язковому використанні захисних окулярів.

За сумарною дією на організм найбільшу небезпеку становлять дециметрові хвилі.

При обслуговуванні радіопередавачів технічний персонал часто знаходиться на відстані менше 50 м.

Для захисту від опромінювання використовують екранування приміщень, захисні окуляри, захисні костюми тощо. Скорочується тривалість робочого часу. Не менше одного разу на рік проводиться санітарний нагляд за такими об'єктами, медичний огляд працівників.

При розміщенні потужних радіостанцій необхідно витримувати радіус санітарно-захисних зон, який залежить від потужності передавача, довжини його робочих хвиль, і висоту встановлювання антени.

При розміщенні потужних радіопередавачів встановлюються зони «суворого режиму», в межах яких не повинно бути населених пунктів.

Конструкції будов і споруд послаблюють електромагнітне опромінювання (табл. 2.7).

Таблиця 2.7

ПОСЛАБЛЕННЯ ЕЛЕКТРОМАГНІТНОГО ОПРОМІНЮВАННЯ БУДІВЕЛЬНИМИ КОНСТРУКЦІЯМИ

Конструкція і матеріали	Товщина, см	Послаблення, дБ (для хвиль довжиною λ, см)		
		0,8	3	10
Капітальна цегляна стіна	70	—	21	16
Штукатурена цегляна стіна	15		12	8
Міжповерхове перекриття			22	20
Вікна з подвійними рамами	—	—	18	7
Цегла	12	20	15	15
Штукатурка	1,8	12	8	—
Скло	0,28	2	2	—
Фанера	0,4	2	1	

Матеріали стін і перекрить будівель, фарбовані поверхні не тільки поглинають, а й відбивають електромагнітні хвилі. Масляна фарба відбиває до 30 % енергії сантиметрових хвиль залежно від кута падіння хвиль.

Електровакуумні прилади, які працюють при напрузі більше 5 кВ (генератори, кенотрони, електронно-променеві трубки й ін.), можуть

стати потенціально небезпечними джерелами рентгенівського опромінювання. Рентгенівське випромінювання, яке генерується при напрузі 5—100 кВ, називають м'яким, а при напрузі більше 100 кВ — жорстким. Воно відноситься до іонізуючого випромінювання, яке призводить до змін в організмі. Променеве ураження розвивається не відразу, тривалість появи його наслідків може становити від декількох хвилин до десятків років залежно від дози опромінювання, його характеру і чутливості організму. Наслідки променевого ураження можуть успадковуватися. Доза опромінювання накопичується з часом.

У нашій державі діють санітарні правила роботи з джерелами рентгенівських променів, які суворо регламентують можливості роботи персоналу з тими чи іншими джерелами випромінювання і тривалість такої роботи.

Рівень рентгенівського опромінювання може вимірюватися за допомогою дозиметрів типу ДРГЗ-0,3 («Аргунь»).

Параметри надвисокочастотного опромінювання нормуються за інтенсивністю (щільністю), тривалістю дії і енергетичним навантаженням. Енергетичне навантаження W дорівнює добутку щільності потоку енергії F на тривалість впливу T:

$$W = FT.$$

Для осіб, пов'язаних з роботою на надвисокочастотному обладнанні, рівень щільності потоку в безперервному й імпульсному режимах опромінювання не повинен перевищувати 1000 мкВт/см^2.

При цьому гранично допустимий рівень енергетичного навантаження протягом робочої зміни становить 200 мкВт·год/см^2 при безперервному опромінюванні і 2000 мкВт·год/см^2 в імпульсному режимі для скважності 10 з урахуванням біологічного послаблення організмом цього впливу. Рівень щільності потоку енергії на робочих місцях і місцях можливого перебування людей, не пов'язаних із роботою на цьому обладнанні, не повинен перевищувати 500 мкВт/см^2 в імпульсному режимі, а протягом робочого дня — не більше 1000 мкВт/см^2.

Максимальна тривалість часу (T) перебування людей в зоні опромінювання обчислюється так:

$$T = \frac{W_{\text{г.д.р}}}{F},$$

де $W_{\text{г.д.р}}$ — гранично допустимий рівень енергетичного навантаження.

Наприклад, для людини, професійно пов'язаної з обладнанням, при безперервному опромінюванні з щільністю потоку енергії $F = 50$ мкВт/см^2

$$T = \frac{200}{50} = 4.$$

За одночасного впливу на організм безперервного й імпульсного джерел опромінювання, наприклад при декількох джерелах опромінювання, сумарне енергетичне навантаження на організм

$$W_c = \sum_{i=1}^{n_1} W_{i\text{б}} + \sum_{i=1}^{n_2} k_i W_{i\text{імп}},$$

де n_1, n_2 — кількість джерел випромінювання відповідно у безперервному та імпульсному режимах роботи; k_i — коефіцієнт, який враховує послаблення біологічного впливу при імпульсному характері опромінювання з урахуванням скважності; $W_{\text{б}}$, $W_{\text{мп}}$ — енергетичне опромінення з урахуванням скважності; $W_{i\text{б}}$, $W_{i\text{ імп}}$ — енергетичне навантаження організму під впливом i-го джерела безперервного й імпульсного впливу відповідно.

Металеві елементи конструкції апаратів, які можуть опинитися під напругою внаслідок порушення ізоляції, повинні бути електрично з'єднані між собою і заземлені. Для захисного заземлення можна використати як штучне, так і природне заземлення. Для штучного заземлення використовуються сталеві вертикально закладені в землю труби діаметром 3—5 см зі стінками завтовшки більше 4 мм і завдовжки більше 3 м або металеві стержні діаметром 10—12 мм, завдовжки біля 10 м. Для штучного заземлення в агресивних ґрунтах (кислих чи лужних) застосовується мідь або оцинкована сталь. Не можна для цих цілей використовувати алюмінієві оболонки кабелів, а також алюмінієві дроти, тому що в ґрунті вони окислюються і не мають при цьому необхідної струмопровідності.

Природним заземленням можуть бути зроблені і прокладені в землі водопровідні мережі й інші металеві трубопроводи, металеві конструкції й арматура залізобетонних виробів, які мають хороший контакт з ґрунтом, свинцеві оболонки кабелів, прокладених у землі.

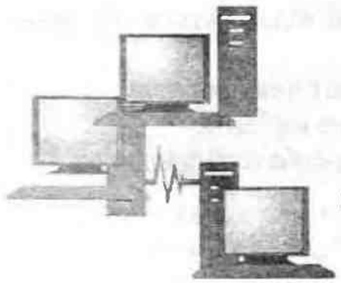
Категорично забороняється використання для цього трубопроводів горючих речовин і газів, труб опалення, каналізації і відводу блискавки. Заземлювальні проводи, шини і стержні, прокладені в приміщенні, мають бути доступні для огляду і мати захист від механічних пошкоджень і механічного впливу. З'єднання елементів заземлення виконується шляхом зварювання, а в разі з'єднання зазем-

лювальних дротів із водопровідною трубою в місці з'єднання труба має бути зачищена від фарби та іржі й обкладена в цьому місці свинцевою або алюмінієвою пластиною.

Опір заземлювального пристрою відносно ґрунту має бути не більше 4 Ом. Цей опір контролюється не менш одного разу на рік.

Питання для самоперевірки

1. Які шкідливі і небезпечні фактори впливають на користувачів та інженерно-технічний персонал під час технічного обслуговування КС та М?
2. Які заходи електробезпеки необхідно виконувати на об'єктах КС та М?
3. Які правила протипожежної безпеки необхідно виконувати при експлуатації засобів КС та М?



Лабораторні роботи

Загальні методичні вказівки

Лабораторні роботи виконуються згідно з програмою курсу «Експлуатація комп'ютерних систем та мереж» для спеціалістів та магістрів спеціальності 8.0915001 «Комп'ютерні системи та мережі» задля напрацювання у студентів практичних навичок і закріплення теоретичних знань з питань технічної експлуатації електронних обчислювальних машин (ЕОМ), систем та мереж, а також дослідження принципів побудови й використання апаратно-програмних засобів підвищення надійності та підтримки експлуатаційного обслуговування систем автоматичного контролю, діагностики, відновлення.

Курс «Експлуатація комп'ютерних систем та мереж» розроблено відповідно до вимог кредитно-модульної системи оцінки знань і складається з двох модулів, що передбачає проведення чотирьох лабораторних робіт у кожному модулі.

Загальною метою лабораторних робіт є поглиблення та закріплення знань з розділів курсу «Експлуатація комп'ютерних систем та мереж». Відповідно до загальної структури посібника лабораторні роботи поділено на два модулі — «Системи контролю функціонування комп'ютерних систем та мереж» та «Процеси експлуатаційного обслуговування комп'ютерних систем та мереж».

У процесі виконання лабораторних робіт студенти ознайомлюються з основами кодування інформації в ЕОМ, засвоюють основні методи підвищення достовірності передачі інформації в ЕОМ, основні принципи побудови перешкодозахисних кодів, а також набувають практичних навиків проектування схем вбудованого контролю, настроювання мережного інтерфейсу *Windows* XP при підключенні в локальну мережу. Студенти ознайомлюються зі спеціалізованим програмним забезпеченням для роботи в локальних комп'ютерних мережах, яке використовується при зборі інформації при мережних настроюваннях комп'ютерів та пошуку інформації на мережних ресурсах, при обміні текстовими повідомленнями між користувачами, при адмініструванні мережних ресурсів комп'ютера.

На виконання кожної лабораторної роботи відводиться дві академічні години. За цей час студент повинен:

— підготувати протокол звіту для даної лабораторної роботи;
— одержати у викладача індивідуальний номер варіанта;
— відповідно до номера варіанта виконати передбачене завдання;
— зробити висновки;
— переглянути відповідну програмну модель;
— відповісти на контрольні запитання.

Звіт про виконання лабораторної роботи має містити:

1. Титульну сторінку.
2. Мету роботи.
3. Короткі теоретичні відомості.
4. Порядок виконання лабораторної роботи.
5. Основні висновки.

До оформлення звіту висуваються такі вимоги:

1. Робота оформлюється на аркушах паперу форматом А4 або в окремому зошиті з лабораторних робіт.
2. На титульній сторінці мають бути вказані:
   - тема лабораторної роботи;
   - назва дисципліни;
   - номер варіанта;
   - ким виконана робота (ПІБ, номер групи, факультет);
   - ким прийнята (ПІБ).
3. У коротких теорстичних відомостях викладаються основні положення даної теми.
4. Хід роботи має містити:
   - вихідні дані;
   - етапи їх перетворення;
   - отриманий результат;
   - знімки з екрана ПК;
   - висновки про результати виконання роботи.

ЛАБОРАТОРНА РОБОТА № 1

Дослідження методів контролю передачі інформації в комп'ютерних системах та мережах

Мета роботи:

ознайомитися з методами і засобами контролю вірогідності переданої інформації та набути практичні навички у формуванні контрольних розрядів.

Короткі теоретичні відомості

Контроль вірогідності переданої інформації — це перевірка інформації на наявність визначених помилок, що можуть виникнути в процесі її передачі.

Контроль передачі інформації може здійснюватися за допомогою апаратних і програмних засобів.

Передача інформації може протікати як у просторі (по каналах зв'язку), так і в часі. Під передачею в часі мається на увазі збереження інформації.

Операції пересилання інформації між різними пристроями ЕОМ виконують частіше за інші операції, на них витрачається від 80 до 90 % машинного часу. Тому особливу увагу варто приділяти контролю вірогідності інформації, переданої каналами зв'язку.

Розглянемо апаратні засоби контролю, застосовувані при передачі інформації в просторі.

Контроль інформації, переданої каналами зв'язку, здійснюється за допомогою спеціалізованих методів контролю.

Усі методи можна розділити на три основні групи:

1) метод дублювання;
2) метод використання мажоритарних систем;
3) методи використання спеціальних кодів.

До третьої групи належать: метод формування контрольних розрядів (метод контролю пріоритету), побудовані на його основі простий і модифікований коди Хемінга, метод групового кодування, циклічний код і багато інших. Усі перелічені коди є надлишковими.

Кодова надмірність — це кількість додаткових розрядів, використовуваних понад мінімуму, необхідного для представлення якої-небудь інформації.

Кількість розрядів, якою відрізняється одна кодова комбінація від іншої, називається кодовою відстанню. Розрізняють мінімальну і максимальну кодові відстані. Мінімальна кодова відстань вказує на кількість розрядів, якою відрізняється одна дозволена кодова комбінація від іншої.

Мінімальна кодова відстань для простих не надлишкових кодів завжди дорівнює одиниці. При цьому виявити і скоригувати помилки неможливо. Прикладом коду з мінімальною кодовою відстанню, рівною 2, є код, що містить один контрольний розряд. Для переведення однієї дозволеної кодової комбінації в іншу необхідно внести зміни в два розряди — один інформаційний і один контрольний.

За призначенням всі методи контролю поділяються на такі, що виявляють перекручування, і такі, що коригують кодове перетво-

рення (коригувальні). Мінімальна кодова відстань характеризує здатності коду як виявляти, так і коригувати.

$$d_{\min} \geq t + 1, \qquad (Л.1.1)$$

де t — кількість виявлених помилок,

$$d_{\min} \geq 2p + 1, \qquad (Л.1.2)$$

де p — кількість коректованих помилок.

Згідно зі статистикою найчастіше трапляються одиничні помилки. Вони виникають у 95—96 % випадків. Тому розглянемо методи виявлення і корекції одиничних помилок.

Метод дублювання інформації

Суть даного методу полягає в дублюванні переданої інформації з наступним зіставленням результатів передачі. При використанні методу дублювання інформації кількість вхідних регістрів приймального пристрою збільшується до двох.

Результатом роботи схеми порівняння за відсутності перекручувань є вироблення дозвільного сигналу низького рівня. Саме цей сигнал дозволяє здійснити передачу інформації із одного з вхідних регістрів приймача до адресованого елемента пристрою. У випадку виявлення перекручування схема порівняння робить сигнал помилки (сигнал високого рівня), що свідчить про необхідність повторної передачі інформаційного слова.

Схема порозрядного порівняння являє собою схему додавання за модулем 2 (рис. Л.1.1).

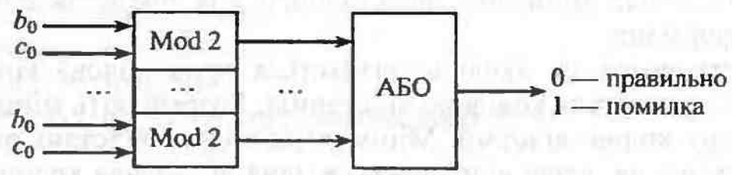


Рис. Л.1.1. Схема порозрядного порівняння

Метод дублювання інформації гранично простий у реалізації, хоча й вимагає введення додаткових апаратних засобів.

Застосування даного методу дає змогу знайти будь-яку кількість помилок, що виникли в результаті передачі. Винятком є перекручування однойменних розрядів. Вони можуть бути викликані, наприклад, порушенням цілісності однієї з вихідних ліній передавального каналу.

Розрізняють кілька видів методу дублювання:

— метод дублювання передачі інформації;

— метод дублювання каналів передачі.

І в тому, і в іншому випадку необхідні додаткові часові витрати на повторну передачу перекрученого слова.

Методи використання мажоритарних систем

Ці методи засновані на зіставленні однойменних розрядів інформаційного слова і на ухваленні рішення про його значення голосуванням за більшістю (фр. *majorité* — більшість).

При цьому число вхідних регістрів приймального пристрою має бути непарним. Найпростішим представником цієї групи методів є метод троювання інформації. Структурна схема методу троювання наведена на рис. Л.1.2.

Тут кількість вхідних регістрів приймача збільшується до трьох. Дані передаються тричі, при цьому на першому такті інформація надходить у регістр B, на другому — у регістр C, на третьому — у регістр D.

Рис. Л.1.2. Схема апаратної реалізації методу троювання

Наявність мажоритарної схеми дає змогу не тільки знайти будь-яку кількість помилок (окрім помилок, що виникають в однойменних розрядах), а й сформувати коректне інформаційне слово. Приклад побудови мажоритарної схеми для одного з розрядів представлено на рис. Л.1.3.

Цей метод досить простий у реалізації, однак вимагає великих апаратних витрат порівняно з методом дублювання.

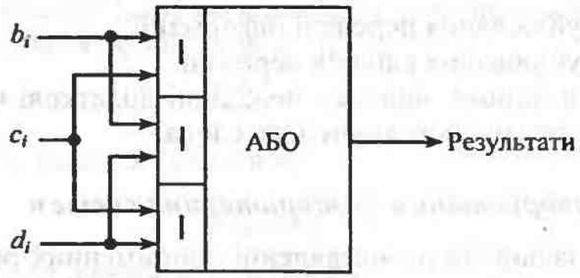


Рис. Л.1.3. Один розряд мажоритарної схеми

При використанні мажоритарних схем контролю у разі виявлення помилки необхідно передбачити автоматичне відключення двох каналів — одного справного й одного несправного. Однак подібну операцію можна провести лише тоді, коли вихідна система містить п'ять і більше каналів зв'язку.

Метод формування контрольних розрядів

Метод формування контрольних розрядів є основним у теорії кодування інформації. У літературі його часто називають методом контролю біта паритету (англ. *parity* — рівність, парність). Він полягає в підрахунку кількості одиниць вихідного (не надлишкового) коду і формуванні відповідного цьому коду значення контрольного (надлишкового) розряду.

Існує два види даного методу:

— формування контрольного розряду з доповненням до парності;

— формування контрольного розряду з доповненням до непарності.

За парної кількості одиниць коду значення контрольного розряду буде нульовим, а за непарної — одиничним.

Легко встановити зв'язок між кодуванням при контролі парності з використанням додавання за модулем 2. Якщо кількість одиниць у слові має бути парною, то в контрольний розряд записується прямий код суми за модулем 2 усіх інформаційних розрядів слова. При контролі на непарність у розряд заноситься зворотне значення зазначеної суми (рис. Л.1.4).

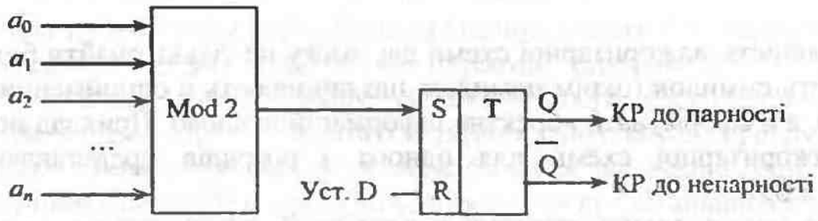


Рис. Л.1.4. Схема формування контрольного розряду

Таким чином, передана по каналах зв'язку кодова комбінація являє собою сукупність інформаційних і одного контрольного розрядів. Для подальшої роботи з отриманим повідомленням приймальний пристрій має виділяти інформаційну частину коду і переконатися в безпомилковості її передачі. Ці функції виконують схеми декодування і контролю.

Схема передачі інформації з використанням методу формування контрольних розрядів запропонована на рис. Л.1.5.

Для інформаційного слова, що знаходиться в регістрі А передавача, обчислюється контрольний розряд. Потім виконується передача слова разом із його контрольним розрядом. Після завершення передачі для прийнятого в регістрі В інформаційного слова приймач формує свій контрольний розряд, що порівнюється із супровідним контрольним розрядом. Збіг значень цих розрядів свідчить про відсутність перекручень (чи про наявність їхнього парного числа). Розбіжність же означає виникнення в процесі передачі одиничної помилки (чи будь-якого іншого непарного числа помилок).

Рис. Л.1.5. Схема реалізації методу контролю паритетів

Код із перевіркою парності має невелику надмірність і не вимагає великих апаратних витрат. Цей код широко застосовується в обчислювальних машинах для контролю передачі між регістрами і контролю зчитуваної інформації в оперативній пам'яті.

Застосування методу контролю паритетів дає змогу знайти будь-яке непарне число помилок, що виникають під час передачі. Однак корекція виявлених перекручень неможлива.

Порядок виконання роботи:

1. Виберіть з табл. Л.1.1 (відповідно до номера отриманого варіанта) вихідне число в шістнадцятковій системі числення.

2. Закодуйте його в двійковій системі числення.

3. Сформуйте для отриманого інформаційного слова два контрольні розряди — один до парності, а другий до непарності.

4. Завантажте в персональний комп'ютер (ПК) моделювальну програму CI.EXE, розташовану в директорії C:\EXPLUAT\LAB1.

5. Ознайомтеся з настановами до її роботи.

6. Перевірте результати вашого кодування, зіставивши їх з результатами роботи програмної моделі.

7. Закріпіть отримані знання, відповівши на контрольні питання.

8. Оформіть протокол.

Таблиця Л.1.1

Номер варіанта	Число, представлене в 16-ковій системі числення
1	10
2	5F
3	3A
4	A1
5	11
6	17
7	7B
8	C3
9	E8
10	99
11	F1
12	35
13	78
14	95
15	5C

Питання для самоперевірки

1. Які методи контролю передачі інформації Вам відомі?

2. Що таке мінімальна кодова відстань і як вона зв'язана з коригувальними здібностями кодів?

3. Що таке мажоритарна система контролю?

4. Чим відрізняються результати роботи схеми дублювання від схеми троювання?

ЛАБОРАТОРНА РОБОТА № 2

Формування коригувального коду Хеммінга

Мета роботи:

ознайомитися з методиками формування простого і посиленого кодів Хеммінга. Здобути практичні навички побудови кодів.

Короткі теоретичні відомості

На основі формування розрядів парності (чи непарності) побудовані різні методи контролю вірогідності переданої інформації, наприклад, коригувальні коди Хеммінга, групові коригувальні коди, циклічні коди та ін.

Коригувальні коди застосовують в ЕОМ для передачі інформації в просторі і часі, в оперативних запам'ятовувальних пристроях. Використання таких кодів дає змогу не тільки виявити, а й виправити помилки.

Простий код Хеммінга. Велике поширення дістав простий коригувальний код Хеммінга. Цей код має мінімальну кодову відстань, рівну трьом ($d_{min} = 3$), що свідчить про його належність до кодів, які дають змогу виявляти до двох помилок і виправляти одиничні помилки.

В основу побудови коду Хеммінга для інформаційних m-розрядних слів покладено метод формування контрольних розрядів парності. Кількість контрольних розрядів відповідає кількості контрольних груп коду Хеммінга і визначається виразом:

$$2^r - r - 1 \geq m, \qquad (Л.2.1)$$

де m — кількість інформаційних розрядів слова; r — кількість контрольних розрядів слова.

Кількість розрядів формованого кодового слова Хеммінга дорівнює сумі інформаційних і контрольних розрядів, тобто $k = m + r$. Усі розряди кодового слова нумеруються від 1 до k, починаючи з правого (молодшого) розряду.

У кожну контрольну групу входять розряди кодового слова, якщо в їхніх двійкових номерах міститься «1», що вказує на належність цих розрядів до визначеної групи. Наприклад, п'ятий розряд входить у першу і другу контрольні групи, а перший, другий і четвертий розряди — тільки в одну з контрольних груп: першу, другу і третю відповідно.

Розряди кодового слова, що входять тільки в одну контрольну групу (1; 2; 4; 8; ...), є контрольними розрядами цих груп.

Розряди, присутні у більш ніж одній контрольній групі, є інформаційними розрядами вихідного слова (3; 5; 6; 7; 9; 10; 11; 12; ...).

Формування кодового слова Хеммінга починається з послідовного запису (біт за бітом) вихідного інформаційного слова у відповідні розряди формованого коду (маючи визначені контрольні розряди, як показано на рис. Л.2.1).

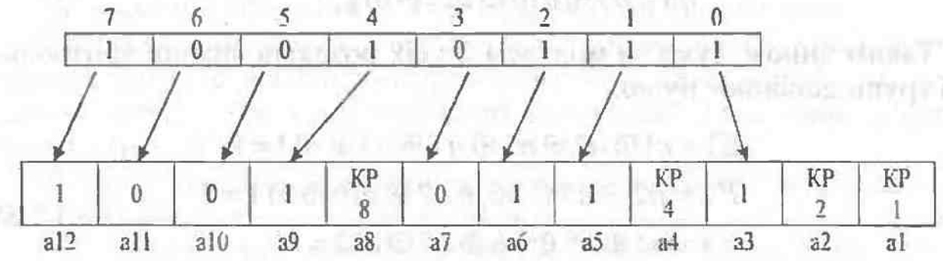


Рис. Л.2.1. Формування простого коду Хеммінга

Розглянемо формування коду Хеммінга для 8-розрядного інформаційного слова. Для цього запишемо номери розрядів кодового слова Хеммінга в десятковій та двійковій системі числення (табл. Л.2.1).

Таблиця Л.2.1

	4-та контрольна група 8, 9, 10, 11, 12,...	3-тя контрольна група 4, 5, 6, 7, 12,...	2-га контрольна група 2, 3, 6, 7, 10, 11,...	1-ша контрольна група 1, 3, 5, 7, 9, 11...
1-	0	0	0	1
2-	0	0	1	0
3-	0	0	1	0
4-	0	1	0	0
5-	0	1	0	1
6-	0	1	1	0
7-	0	1	1	1
8-	1	0	0	0
9-	1	0	0	1
10-	1	0	1	0
11-	1	0	1	1
12-	1	1	0	0

Контрольні розряди кожної контрольної групи визначаються додаванням за модулем 2 інформаційних розрядів відповідної групи:

$$\begin{aligned} a1 &= a3 \oplus a5 \oplus a7 \oplus a9 \oplus a11; \\ a2 &= a3 \oplus a6 \oplus a7 \oplus a10 \oplus a11; \\ a4 &= a5 \oplus a6 \oplus a7 \oplus a12; \\ a8 &= a9 \oplus a10 \oplus a11 \oplus a12. \end{aligned} \qquad (Л.2.2)$$

Таким чином, сума за модулем 2 усіх розрядів кожної контрольної групи дорівнює нулю:

$$\begin{aligned} E1 &= a1 \oplus a3 \oplus a5 \oplus a7 \oplus a9 \oplus a11 = 0; \\ E2 &= a2 \oplus a3 \oplus a6 \oplus a7 \oplus a10 \oplus a11 = 0; \\ E3 &= a4 \oplus a5 \oplus a6 \oplus a7 \oplus a12 = 0; \\ E4 &= a8 \oplus a9 \oplus a10 \oplus a11 \oplus a12 = 0. \end{aligned} \qquad (Л.2.3)$$

Числа $E1$, $E2$, $E3$ і $E4$ утворюють позиційне двійкове коригувальне число $E4\ E3\ E2\ E1$, що при формуванні коду Хеммінга дорівнює 0 0 0 0.

При прийнятті (чи зчитуванні) кодового слова Хеммінга коригувальне число свідчить про безпомилкову передачу чи про наявність помилки і її місце в кодовому числі.

Під час прийняття кодового слова виконується повторне формування коригувального числа. Для цього знов формуються контрольні розряди $a1'$, $a2'$, $a4'$, $a8'$, ..., які зіставляються через додавання за модулем 2 з контрольними розрядами $a1$, $a2$, $a4$, $a8$, ..., що були сформовані передавальним пристроєм:

$$\begin{aligned} E1 &= a1 \oplus a1'; \\ E2 &= a2 \oplus a2'; \\ E3 &= a4 \oplus a4'; \\ E4 &= a8 \oplus a8'. \end{aligned} \qquad (Л.2.4)$$

Після визначення контрольних сум кожної контрольної групи можна записати коригувальне число.

Якщо $E4\ E3\ E2\ E1 = 0\ 0\ 0\ 0$, то це означає, що помилки в кодовому слові не виявлені (тобто помилок немає чи їхня кількість кратна трьом).

У разі, якщо коригувальне число відмінне від нуля, його величина в двійковій системі числення буде дорівнювати порядковому номеру розряду кодового слова Хеммінга, в якому виявлена помилка. Це пояснюється тим, що помилка в деякому розряді кодового слова порушить парність сум у тих контрольних групах, у яких даний розряд присутній.

Наприклад, п'ятий біт кодового слова належить першій і третій контрольним групам. Тому помилка в п'ятому розряді кодового слова приведе до зміни парності цих груп. У цьому випадку $E1 = 1$, $E2 = 0$, $E3 = 1$ і $E4 = 0$, тобто коригувальне число матиме значення $E4\ E3\ E2\ E1 = 0\ 1\ 0\ 1$.

Це значення відповідає п'ятому розряду кодового слова.

Корекція помилки в обчисленому розряді виконується досить просто: поточне значення «дефектного» біта перед наступною передачею інвертується через додавання за модулем 2 зі словом розузгодження (рис. Л.2.2).

У такий спосіб код Хеммінга виявляє свої коригувальні можливості в разі виникнення одиничних помилок.

При виникненні двох помилок у кодовому слові коригувальне число також буде ненульовим. Це буде означати, що в кодовому слові з'явилися помилки. Величина коригувального числа вказува-

тиме на деякий розряд кодового слова Хеммінга. Інвертування цього розряду призведе не до корекції помилок, а до внесення ще однієї, третьої, помилки. Наявні помилки будуть замасковані, «заховані». Кодове слово з двома помилками в разі внесення третьої помилки перейде з «дефектного» стану в новий дозволений стан, однак він не буде відповідати вихідному інформаційному слову. Це пояснюється тим, що помилки в двох розрядах кодового слова компенсують одна одну.

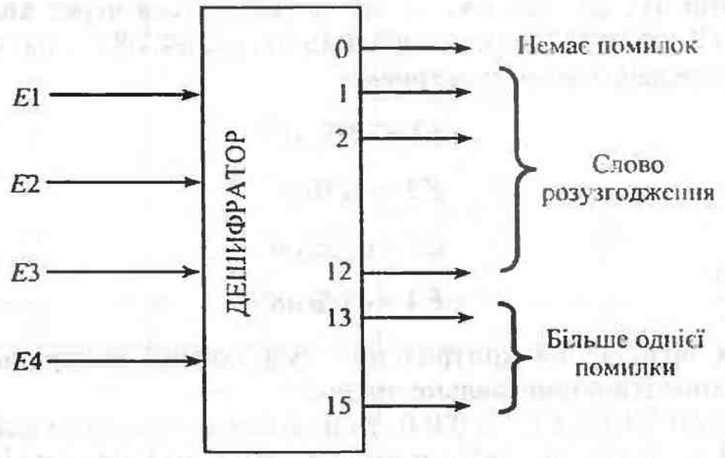


Рис. Л.2.2. Формування слова розузгодження

Розглянемо цю ситуацію на прикладах:

1. Помилка в третьому і п'ятому інформаційних розрядах. Помилка в третьому розряді порушить парність у першій і другій контрольних групах, а помилка в п'ятому розряді — у першій і третій контрольних групах. Дві помилки у першій контрольній групі компенсують одна одну, і парність у цій групі не порушиться ($E1 = 0$). Зберігається одночасне порушення парності в другій і третій контрольних групах ($E2 = 1$ і $E3 = 1$), що буде вказувати нібито на помилку в шостому розряді кодового слова.

2. Помилки в контрольних першому і другому розрядах. Оскільки ці розряди входять тільки у свої контрольні групи, ці дві помилки спричинять порушення парності в цих групах ($E1 = 1$ і $E2 = 1$) і приведуть до рекомендації коригувати третій розряд.

3. Помилки в контрольних четвертому й інформаційному п'ятому розрядах. Помилка п'ятого розряду порушить парність у першій і третій контрольних групах, але помилка в четвертому розряді компенсує помилку п'ятого розряду в третій групі. Порушення парності буде виявлятися тільки в першій контрольній групі ($E1 = 0$), що вказуватиме на необхідність корекції першого розряду.

Посилений код Хеммінга. За ненульовим коригувальним числом простого коду Хеммінга неможливо визначити причину: одинична чи подвійна помилка виникла при передачі інформації. Щоб уникнути такої невизначеності і бути впевненим, що виконується корекція одиничної помилки, а не маскування подвійної помилки, застосовують модифікаційний (посилений) код Хеммінга. Для його формування необхідно для вихідного інформаційного слова створити простий код Хеммінга, а потім додати ще один додатковий розряд — контрольний розряд парності (чи непарності) усього кодового слова Хеммінга. Мінімальна кодова відстань при цьому збільшується до чотирьох ($d_{min} = 4$), але цього недостатньо для корекції двох помилок. Можливість корекції одиничної помилки зберігається, але при виникненні двох помилок з'являється можливість однозначно констатувати цей факт і свідомо прийняти рішення про маскування цих помилок або про відмову приймати таке кодове слово.

Порядок виконання роботи

1. Виберіть з табл. Л.2.2 (відповідно до номера отриманого варіанта) вихідне число в десятковій системі числення.

Таблиця Л.2.2

Номер варіанта	Число	Номер варіанта	Число
1	135	16	115
2	88	17	103
3	203	18	196
4	100	19	229
5	156	20	70
6	164	21	170
7	59	22	210
8	246	23	125
9	106	24	75
10	149	25	240
11	56	26	113
12	201	27	250
13	211	28	171
14	98	29	89
15	188	30	148

2. Закодуйте його в двійковій системі числення.

3. Сформуйте для отриманого інформаційного слова простий і модифікаційний коди Хеммінга.

4. В отримані кодові комбінації внести по черзі одну, а потім дві помилки і відповідно до алгоритму спробувати відкоригувати їх.

5. Завантажте в ПК моделювальну програму HEMMING.EXE, розташовану в директорії C:\EXPLUAT\LAB2.

6. Ознайомтеся з правилами її роботи.

7. Перевірте результати вашого кодування, зіставивши їх з результатами роботи програмної моделі.

8. Оформіть протокол.

Питання для самоперевірки

1. Як формується простий код Хеммінга?

2. Яким чином обчислюється кількість контрольних розрядів простого коду Хеммінга?

3. Як формується модифікований код Хеммінга?

4. Чим відрізняється простий код Хеммінга від модифікаційного?

5. Чому дорівнює мінімальна кодова відстань простого коду Хеммінга і про що вона свідчить?

ЛАБОРАТОРНА РОБОТА № 3

Дослідження методів групового кодування інформації

Мета роботи:

ознайомитися з методикою групового кодування і здобути практичні навички формування контрольних слів.

Короткі теоретичні відомості

Метод групового кодування найчастіше використовується під час записування та зчитування інформації з носіїв.

В основу даного методу покладено принцип формування контрольних розрядів.

Метод групового кодування потребує блочного передавання даних, тому його не можна застосувати для одиничних передач. При цьому контрольні розряди формуються в двох напрямах: побайтно (для кожного інформаційного слова блоку формується свій контрольний розряд) та порозрядно (для всіх однойменних розрядів блоку формуються свої контрольні розряди). Слово побайтного контролю називається словом поздовжнього контролю, а слово порозрядного — байтом поздовжнього контролю (БПК) чи словом вертикального контролю.

Для більш зручного алгоритму виявлення та корекції помилок в інформації представимо фізичний запис у вигляді матриці, в якій стовпці — це інформаційні байти і БПК з контрольними розрядами непарних чисел, а рядки — відповідні розряди байтів, що записуються (табл. Л.3.1).

Таблиця Л.3.1

Кількість розрядів	1	2	3	4	...	М	БПК
0-й	1	0	1	0		0	0
1-й	0	1	1	0		0	0
2-й	0	0	0	1		0	1
3-й	0	0	0	0		0	0
4-й	0	0	0	0		0	0
5-й	0	0	0	0		0	0
6-й	0	0	0	0		0	0
7-й	0	0	0	0		0	0
...							
n	0	0	0	0		0	0
КР	1	1	0	1		0	

З цієї таблиці видно, що одиничні помилки, які з'являються під час передавання інформації (або будь-якого непарного числа перекручень в одному зі слів інформаційного блоку) ведуть до зміни слів поздовжнього й вертикального контролю, що формуються при декодуванні коду з метою контролю. При цьому номер контрольного розряду (КР), що не збігається, вкаже на адресу дефектного слова, а слово розбіжності, сформоване схемою порівняння БПК, дозволить вказати номери перекручених розрядів.

Контролер накопичувача на гнучких магнітних дисках (НГМД)

Для кожного інформаційного слова завдовжки в один, два чи три байти, що надходять паралельним кодом по шинах даних в контролер НГМД для запису на магнітний диск, в контролері виникає КР непарності.

Ці контрольні розряди записуються на носій після кожної групи (байта, двох або чотирьох) інформаційних бітів.

Крім того, в результаті виконання на вхідному інформаційному регістрі контролера додавання за модулем 2 усіх записуваних інформаційних слів формується байт поздовжнього контролю даного фізичного запису. Кожний з розрядів цього байта доповнює до парного числа кількість одиниць одновимірних розрядів усіх записуваних інформаційних слів. Після запису усіх інформаційних слів на носій записується також байт поздовжнього контролю зі своїм контрольним розрядом непарності.

Як приклад розглянемо запис на ГМД шістьох інформаційних блоків:

1-й байт: 10010011

2-й байт: 11111111

3-й байт: 10100001

4-й байт: 00000000

5-й байт: 11001010

6-й байт: 01110011

Для прикладу запису поданої інформації у контролері НГМД (рис. Л.3.1) для кожного байта буде вирахувано контрольний розряд, що доповнює число одиниць інформаційного байта до парного: 1; 1; 0; 1; 1; 0 відповідно. Контрольні розряди додаються до відповідних байтів як 9-й біт і разом записуються на носій інформації. Крім того, у схемі формування контрольного біта — БПК накопичується сума за модулем 2 записуваних інформаційних байтів. Кожен БПК при цьому доповнює однойменні біти усіх інформаційних байтів до парної кількості. Для БПК також визначається контрольний розряд до непарності.

Отже, на ГМД у вигляді одного фізичного запису буде записано блок дев'ятирозрядних слів:

1-ше слово: 100100111

2-ге слово: 111111111

3-тє слово: 101000010

4-те слово: 000000001

5-те слово: 110010101

6-те слово: 011100110

7-ме слово: 011101001

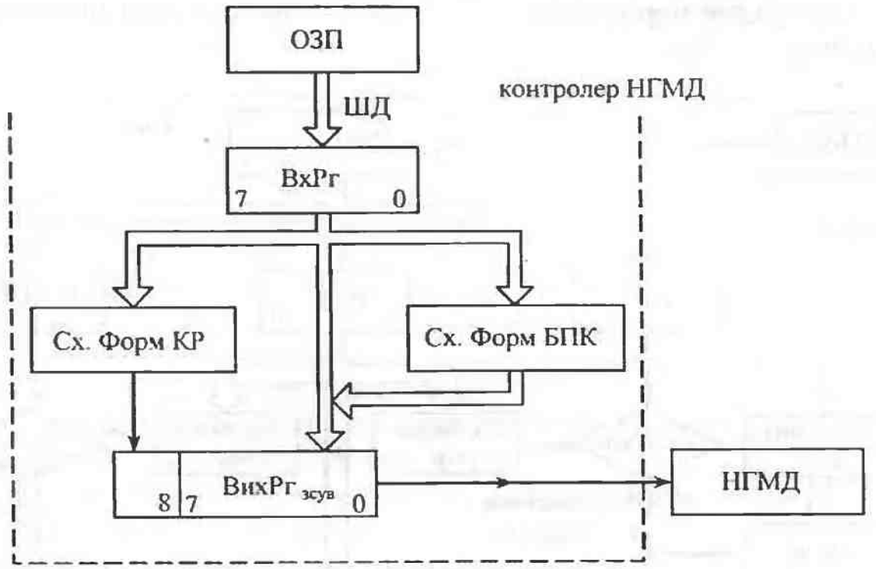


Рис. Л.3.1. Схема запису інформації на НГМД

Послідовне побітове розміщення інформації на магнітній ГМД показано на рис. Л.3.2.

1-й байт 2-й байт 3-й байт	6-й байт	БПК
100100111111111111010001000.....	01110011	0011101001
<----------Фізичний запис---------->		

Рис. Л.3.2. Розміщення записаної інформації на ГМД

При читанні фізичного запису (рис. Л.3.3) контроль інформації виконується за таким алгоритмом.

Кожен 1-й байт, що читається разом із його контрольним розрядом, надходить до контролера НГМД, в якому виконуються такі процедури:

— виділяються інформаційні біти та контрольний біт КР1;

— інформаційні біти передаються в оперативний запам'ятовувальний пристрій (ОЗП), на схему формування БПК2 та схему обчислення контрольного біта читаного інформаційного байта КР2;

— порівнюються контрольні біти: КР1, не читаний з носія, з КР2, обчисленим для читаного байта;

— в разі незбігу значення КР1 та КР2, що свідчить про виявлення помилки в цьому читаному байті, запам'ятовується порядковий номер «дефектного» байта. Прочитаний з носія БПК1 передається на порозрядне порівняння з БПК2, який був сформований під час читання.

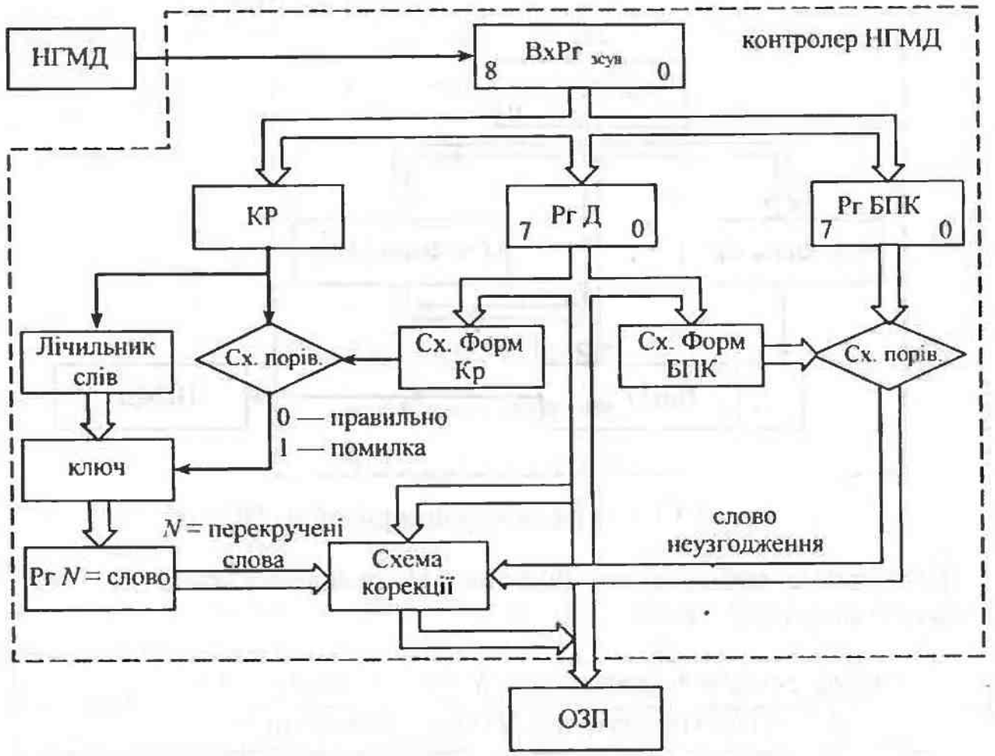


Рис. Л.3.3. Схема зчитування інформації на НГМД

При виявленні одиничної помилки в інформаційному байті КР непарності цього байта визначає номер байта, в якому з'явилася ця помилка (у даному прикладі — це 3-й байт). Розряд БПК (4-й) буде визначати номер помилкового біта. При повторному читанні у момент читання дефективного байта з фіксованим номером визначений розряд цього байта буде відкоригований простим інвертуванням його значення.

У мікропроцесор (МП) буде передано виправлений байт, але на носії помилка збережеться. У подальшому є можливість лише програмними методами виконати перезапис усієї інформації.

У разі виникнення помилки в КР одного з інформаційних байтів (наприклад, у КР 5-го байта) в лічильнику зчитаних байтів буде за-

фіксовано номер цього «дефектного» байта. Але при перевірці БПК та його КР помилка не визначається. Це свідчить, що помилка — в контрольному розряді 5-го інформаційного байта. Оскільки операція читання не переривалася та усі інформаційні байти були записані в ОЗП, сигнали про помилку не робилися та повторне читання не виконувалося.

Якщо помилка виявилася в одному з бітів БПК, лічильник байтів вкаже, що це не інформаційний, а контрольний байт, його контрольний розряд непарності вкаже на виділення в ньому одиничної помилки, а порозрядне порівняння зчитаного БПК з вирахуваним може вказати на «дефектний» біт.

Виявлення помилки у контрольному розряді непарності БПК виконується після зчитування БПК з носія і порівняння його з БПК, вирахуваним під час зчитування.

Порозрядне порівняння цих байтів не виявить неузгоджень в «інформаційних» бітах байтів, що свідчить про одиничну помилку в КР непарності, яку дозволено ігнорувати.

У результаті проведеного аналізу можна зазначити такі особливості даного методу (алгоритму):

- за появи одиничних помилок в будь-якому біті фізичного запису (інформаційних байтах, БПК або контрольних розрядах непарності) ці помилки будуть виявлені і в разі необхідності виправлені;
- за появи більш ніж однієї помилки, але обов'язково непарної кількості в одному байті, контрольний розряд непарності цього байта допоможе зафіксувати його номер. Зчитуючи БПК та виконуючи порозрядне порівняння його зчитаного значення з вирахуваним (КР не сигналізує про порушення парності), можна вказати «дефектні» біти «дефектного байта»;
- у разі появи подвійних (загалом парних) помилок в одному з інформаційних байтів, коли КР непарності не дозволяє зафіксувати «дефектний» байт, ознакою виявлення помилок буде незбіг читаного БПК із тим, який був вирахуваний під час зчитування інформаційних байтів. Під час порозрядного порівняння байтів є можливість виділити номери «дефектних» бітів;
- за появи великої кількості помилок у різних байтах цей факт буде зареєстровано, але можливості їх локалізації та корекції немає, тоді видається повідомлення про ситуацію, що склалася, й з'являється можливість вибору альтернативних варіантів: повторення виконання усієї операції, пропуск цього фізичного запису або запису в ОЗП з ігноруванням помилок.

Розглянемо реалізацію даного алгоритму виконання зчитування інформації з ГМД:

■ здійснюється побітове зчитування інформації з ГМД. Зчитувані байти (2 або 4 байти) зі своїм КР1 непарності (біт вертикального контролю) заносяться у зсувний регістр (ЗР);

■ зчитаний байт паралельним кодом заноситься до вихідного регістру для подання інформації в МП та для вирахування БПК. Вираховується КР2 непарності інформаційних бітів. У лічильник зчитаних байтів заноситься одиниця для підрахунку кількості байтів;

■ порівнюються контрольні розряди непарності: зчитаний КР1 та вирахуваний при читанні КР2. Якщо їх значення збігаються (TRUE = = істина), то зчитаний байт без КР передається на шини даних інтерфейсу. Якщо значення КР не збігаються (FALSE = хибно), то фіксується значення лічильника зчитаних байтів (при подальшому зчитуванні його значення не змінюється);

■ читання інформації триває до читання БПК.

Порядок виконання роботи

1. Виберіть із табл. Л.3.2 (відповідно до номера отриманого варіанта) блок даних, представлений в десятковій системі числення.

Таблиця Л.3.2

№ варіанта	Дані	№ варіанта	Дані
1	1,135,15,20,75	16	5,7,201,17,113
2	103,88,4,3,16	17	16,229,1,188,2
3	16,1,210,5,70	18	149,0,106,1,15
4	240,32,67,56,1	19	75,196,12,3,11
5	2,64,211,164,0	20	64,15,16,59,211
6	9,188,13,56,4	21	103,11,12,164,0
7	24,11,164,211,3	22	33,18,170,0,17
8	0,17,8,246,98	23	125,210,16,3,34
9	148,89,64,20,4	24	15,8,196,70,2
10	5,15,171,18,75	25	4,115,20,98,1
11	203,0,13,1,106	26	21,0,56,240,15
12	100,20,1,3,88	27	113,171,0,0,1
13	16,20,149,64,56	28	15,0,1,229,56
14	59,246,16,33,1	29	246,16,8,100,13
15	0,115,103,4,21	30	75,1,21,115

2. Закодуйте його в двійковій системі числення.
3. Сформуйте слово вертикального та поздовжнього контролю.
4. Завантажте в ПК моделювальну програму.
5. Ознайомтеся з настановами до її роботи.
6. Порівняйте отримані результати з результатами роботи програмної моделі.
7. Закріпіть отримані знання, відповівши на контрольні питання.
8. Оформіть протокол.

Питання для самоперевірки

1. Які помилки не виявляються при застосуванні схеми групового кодування?
2. Поясніть принцип формування БПК.
3. Опишіть принцип роботи контролера НГМД у режимі зчитування інформації з магнітного носія.
4. Опишіть принцип роботи контролера НГМД в режимі записування інформації на магнітний носій.

ЛАБОРАТОРНА РОБОТА № 4

Дослідження методів контролю арифметичних і логічних операцій

Мета роботи:

ознайомитися з методикою основними методами контролю арифметичних і логічних операцій та їх застосуванням.

Короткі теоретичні відомості

Арифметичні операції можна представити у вигляді послідовності таких елементарних операцій: передача слова й операції перетворення вмісту тригерних регістрів — зсув, формування оберненого коду і додавання.

Контроль операції зсуву

Операція зсуву інформації в регістрі являє собою передачу інформації — із i-х розрядів регістрів в $(i - m)$-ні чи $(i + m)$-ні розряди залежно від напряму зсуву (m — число розрядів, на яке виконується зсув). Тому для контролю операції зсуву можна використовувати ті

самі методи, що й для контролю передачі інформації, наприклад контроль парності суми одиниць.

Регістр, в якому виконується зсув, повинен мати додатковий контрольний розряд, що встановлюється перед зсувом у такий стан, щоб сума одиниць регістру разом із контрольним розрядом була парною. Окрім схеми визначення загальної парності, необхідні схеми, що встановлюють парність різниці між кількістю одиниць, що висуваються з регістру, і кількістю одиниць, що надходять до регістру (рис. Л.4.1).

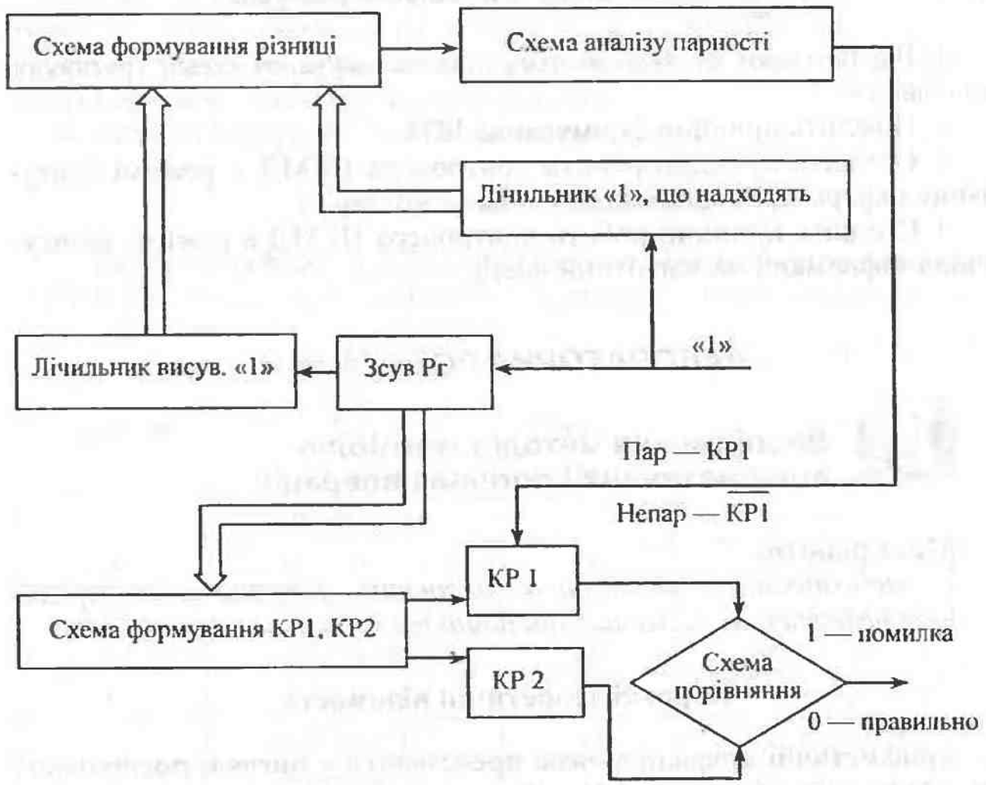


Рис. Л.4.1. Контроль роботи регістру зсуву із заповненням «1»

Якщо у розряди, які звільняються при зсуві, надходять «0» (рис. Л.4.2), то достатньо мати схему для визначення парності одиниць висувних розрядів. Одночасно зі зсувом виконується контрольна операція, що полягає у зміні стану контрольного розряду на протилежне значення при непарності суми одиниць висувних розрядів. Тоді при правильному виконанні зсуву загальна парність суми одиниць в регістрі після зсуву не змінюється.

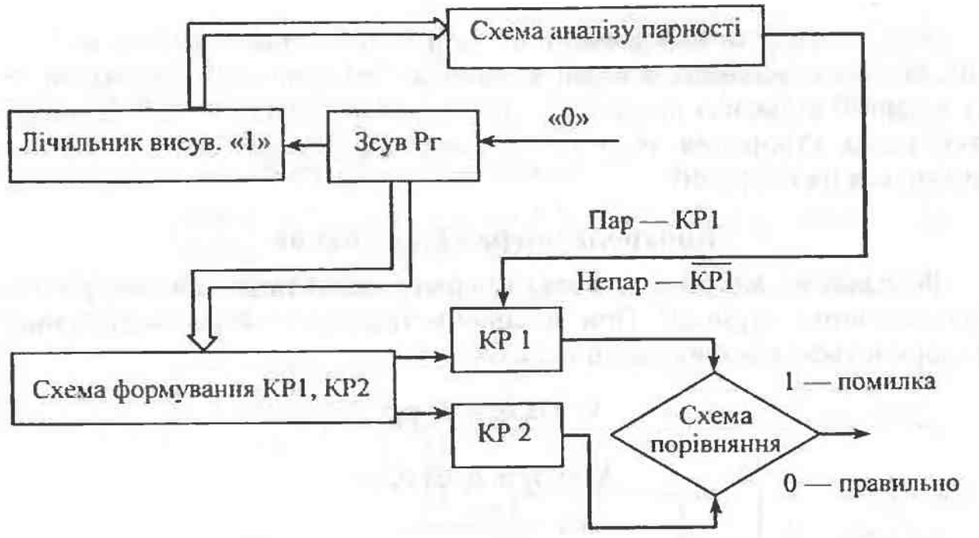


Рис. Л.4.2. Контроль роботи регістру зсуву із заповненням «0»

Контроль операції формування оберненого коду

Операція формування оберненого коду може бути також проконтрольована через використання кодів з перевіркою парності (рис. Л.4.3). Якщо кількість інформаційних розрядів у слові парна, то кількість одиниць в слові парна за парної кількості нулів — 0, і кількість одиниць непарна за непарної кількості нулів — 0. В цьому разі після утворення оберненого коду парність кількості одиниць в слові збережеться, і правильність виконання операції можна визначити, перевіривши збереження парності чи непарності суми одиниць в слові, включаючи контрольний розряд.

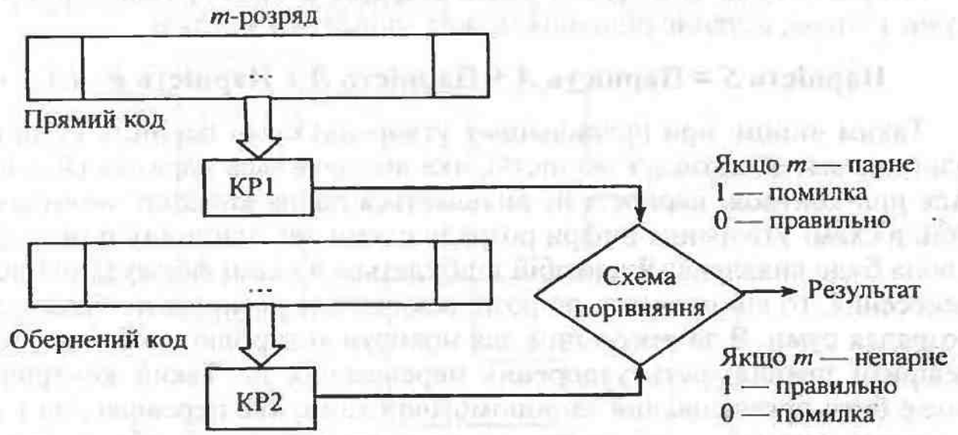


Рис. Л.4.3. Контроль операції формування оберненого коду

Якщо кількість інформаційних розрядів в слові непарна, то парній кількості одиниць в слові відповідає непарна кількість нулів — 0, а парній кількості одиниць — парна кількість нулів — 0. В цьому разі після утворення зворотного коду парність кількості одиниць зміниться на обернену.

Контроль операції додавання

Розглянемо метод контролю операції додавання, що використовує перевірку парності. При додаванні чисел A та B розряди суми S утворюються відповідно до виразів:

$$S_1 = a_1 \oplus b_1 \oplus p_1;$$

$$S_2 = a_2 \oplus b_2 \oplus p_2;$$

$$\ldots\ldots\ldots\ldots$$

$$S_n = a_n \oplus b_n \oplus p_n,$$

де S_i, a_i, b_i, p_i ($i = 1, 2,\ldots n$) — відповідно значення розрядів суми, доданків та перенесення, яке надходить в i-й розряд. Знак $\oplus$ означає додавання за модулем 2; n — кількість розрядів доданків і суми.

Після додавання всіх n рівностей, що були наведені раніше, за модулем 2 отримаємо:

$$S_1 + S_2 + \ldots + S_n = (a_1 + a_2 + \ldots + a_n) +$$

$$+ (b_1 + b_2 + \ldots + b_n) + (p_1 + p_2 + \ldots + p_n).$$

Оскільки сума за модулем 2 всіх розрядів слова виражає парність суми 1 слова, останнє рівняння можна записати у вигляді:

Парність S = Парність A + Парність B + Парність P. (Л.4.1)

Таким чином, при правильному утворенні суми парність суми її одиниць має збігатися з парністю, яка визначається виразом (Л.4.1). Але при контролі парності не виявляється парна кількість помилок. Збій в схемі утворення цифри розряду схеми дає одиничну помилку, і вона буде виявлена. Якщо збій відбудеться в схемі формування перенесення, то він призведе до розповсюдження помилки по багатьох розрядах суми. В зв'язку з цим для повноти контролю необхідно перевірити правильність утворення перенесення p_i. Такий контроль може бути організований за допомогою схеми, яка перевіряє, чи є в кожному розряді — або перенесення у прямому коді, або інверсія перенесення і чи наявне одночасно і те, й інше.

Для перенесень і окремо для суми формуються контрольні розряди парності. Потім схема перевірки парності перевіряє виконання умови (Л.4.1).

Контроль комбінаційних схем

Для контролю комбінаційних схем використовуються два методи.

Перший метод базується на дублюванні виконання операцій на комбінаційних схемах за рахунок подвоєння паралельно працюючих схем (рис. Л.4.4).

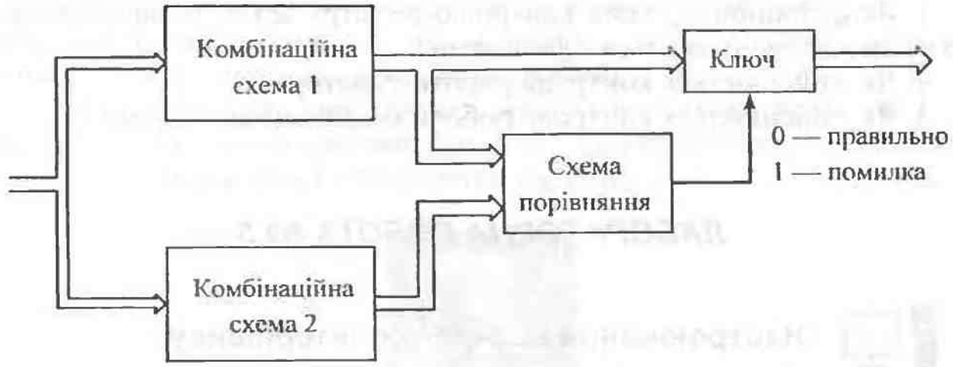


Рис. Л.4.4. Контроль роботи комбінаційних схем методом дублювання

Другий метод перевірки комбінаційних схем базується на знанні закону функціонування цієї схеми (на знанні функції, яка реалізується). Наприклад, для дешифратора, де значення «1» має прийматися тільки в одному розряді і, відповідно, якщо виникла помилка і з'явилися два одиничні сигнали, її можна виявити (рис. Л.4.5).

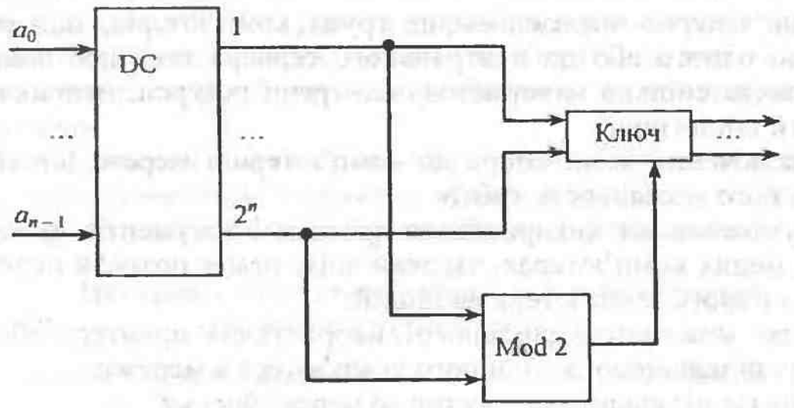


Рис. Л.4.5. Контроль роботи дешифратора

Порядок виконання роботи

1. Завантажте в ПК моделювальну програму.
2. Закріпіть отримані знання за допомогою програмної моделі.
3. Закріпіть знання, відповівши на контрольні питання.
4. Оформіть протокол.

Питання для самоперевірки

1. Як функціонує схема контролю регістру зсуву із заповненням розрядів, що звільняються одиницями?
2. Як здійснюється контроль роботи суматора?
3. Як здійспюється контроль роботи комбінаційних схем?

ЛАБОРАТОРНА РОБОТА № 5

Настроювання мережного інтерфейсу *WINDOWS* XP

Мета роботи:

ознайомитися і вивчення методів настроювання мережного інтерфейсу ***Windows*** **XP** *для підключення в локальну мережу.*

Короткі теоретичні відомості

Комп'ютерна мережа — це група комп'ютерів, підключених один до одного або до центрального сервера так, щоб вони мали можливість спільно використовувати різні ресурси, наприклад, документи і принтери.

Підключення комп'ютера до комп'ютерної мережі істотно розширює його можливості, тобто:

— уможливлює використання програм і документів, розташованих на інших комп'ютерах, завдяки чому немає потреби переносити файли з одного комп'ютера на інший;

— дає можливість спільного використання принтера або факс-модему, підключеного до іншого комп'ютера в мережі:

— надає підключення і доступ до мережі *Internet.*

Основні поняття і визначення

Мережний ресурс — пристрій, до якого існує доступ в комп'ютерній мережі. Це може бути комп'ютер, принтер, папка з даними.

Робоча група — сукупність комп'ютерів, які мають в рамках даної комп'ютерної мережі ім'я, що не повторюється, і схожі права доступу.

Права доступу — сукупність правил доступу до ресурсів комп'ютерної мережі.

Кожен комп'ютер, підключений до комп'ютерної мережі, повинен мати індивідуальне, що не повторюється в рамках даної комп'ютерної мережі, ім'я і входити в якусь робочу групу.

Для доступу до комп'ютерної мережі необхідно звернутися до ярлика **«Сетевое окружение»** (рис. Л.5.1), розташованому на робочому столі. Цей ярлик є посиланням на папку

Рис. Л.5.1. Ярлик **«Сетевое окружение»**

«Сетевое окружение» призначено для організації роботи з комп'ютерною мережею та відображення мережних підключень.

Папка **«Сетевое окружение»** служить для відображення комп'ютерів, які входять в робочу групу, до складу якої входить комп'ютер користувача. При звертанні до нього з'являється вікно **«Сетевое окружение»** (рис. Л.5.2). Для відображення комп'ютерів робочої групи необхідно лівою кнопкою миші на боковій панелі **«Сетевые задачи»** натиснути пункт меню **«Отобразить компьютеры рабочей группы»**.

Для відображення комп'ютерів інших робочих груп та підключення до них треба скористатися пунктом **«Вверх»** на панелі інструментів вікна, що відображає комп'ютери робочої групи користувача.

Настроювання комп'ютера для підключення до комп'ютерпої мережі

Операційна система *Windows* XP одразу після інсталяції має готовий до роботи мережний інтерфейс із усіма необхідними для ро-

боти компонентами. Для підключення комп'ютера до комп'ютерної мережі потрібно тільки вказати ім'я комп'ютера та відповідну робочу групу.

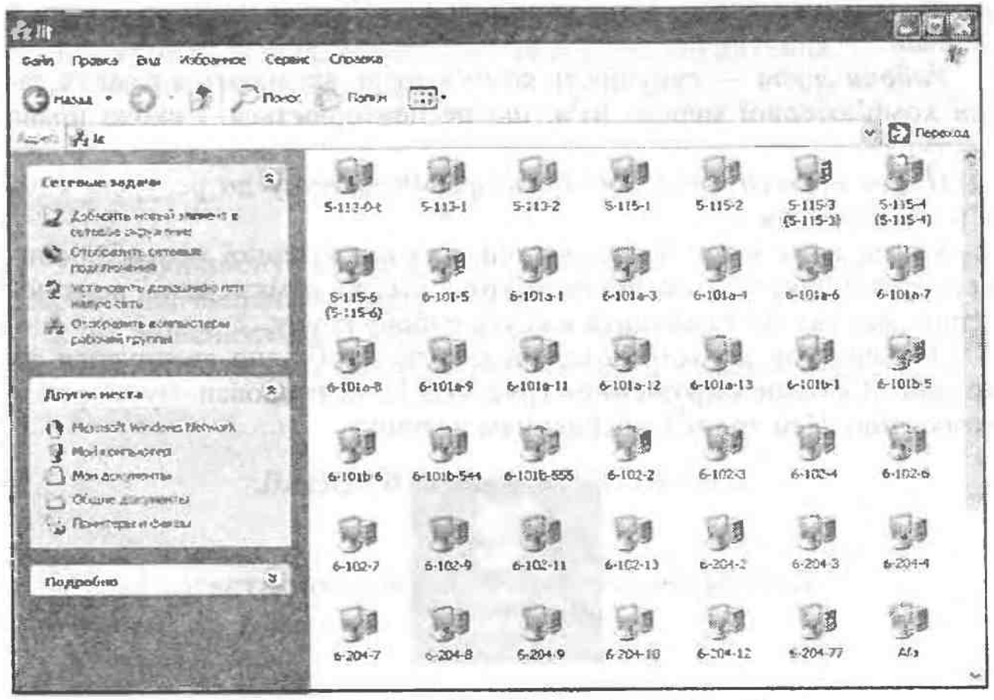


Рис. Л.5.2. Папка «**Сетевое окружение**»

Для цього потрібно натиснути правою кнопкою миші на ярлику «**Мой компьютер**», який розташований на робочому столі, та в меню, що з'явилося, вибрати пункт «**Свойства**». У вікні, що з'явиться, «**Свойства системы**» необхідно перейти на закладку «**Имя компьютера**» (рис. Л.5.3), на якій зазначені параметри, що використовуються для ідентифікації комп'ютера в мережі: «**Полное имя**» — ім'я комп'ютера в мережі; «**Рабочая группа**» — робоча група, до складу якої входить комп'ютер (за замовчуванням — *WORKGROUP*), «**Описание компьютера**» — додаткова ідентифікація комп'ютера в мережі (необов'язковий параметр). Для зміни цих параметрів треба лівою кнопкою миші натиснути пункт «**Изменить...**». При цьому з'явиться вікно «**Изменение имени компьютера**» (рис. Л.5.4), у відповідних полях якого потрібно вказати ім'я комп'ютера та робочої групи, а потім натиснути «**ОК**». Для того, щоб зміни в настроюваннях комп'ютера набули чинності, комп'ютер слід перезавантажити.

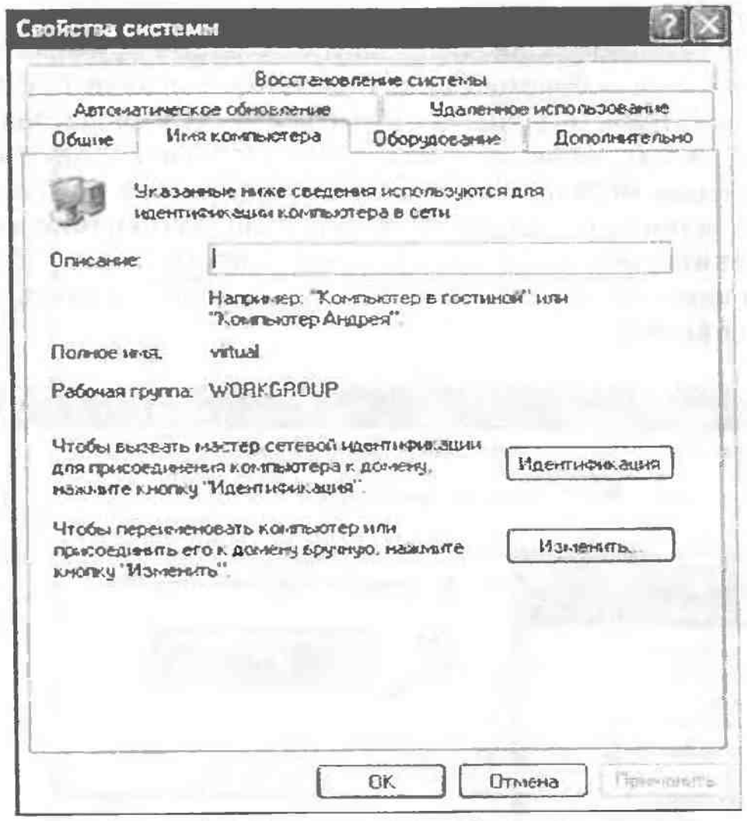


Рис. Л.5.3. Закладка «**Имя компьютера**»

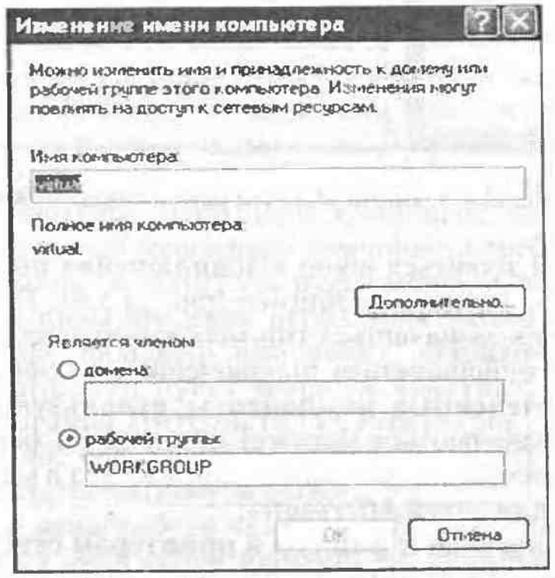


Рис. Л.5.4. Вікно «**Изменение имени компьютера**»

Для зміни мережних настроювань комп'ютера потрібно натиснути правою кнопкою миші на ярлику **«Сетевое окружение»**, який розташований на робочому столі, та в меню, що з'явиться, вибрати пункт **«Свойства»**. У вікні, що з'явиться — **«Сетевые подключения»** (рис. Л.5.5), лівою кнопкою миші необхідно виділити об'єкт **«Подключение по локальной сети»**, після чого на боковій панелі **«Сетевые задачи»** натиснути на пункт меню **«Изменение настроек подключения»** або, натиснувши правою кнопкою миші на об'єкті **«Подключение по локальной сети»**, у меню, що з'явиться, вибрати пункт **«Свойства»**.

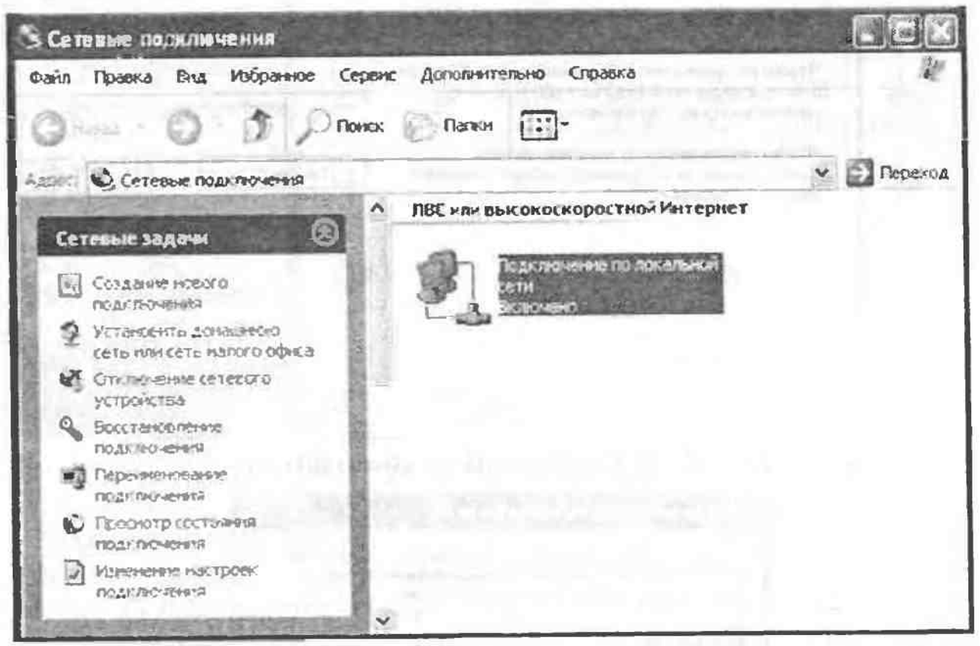


Рис. Л.5.5. Вікно **«Сетевые подключения»**

Після цих дій з'явиться вікно **«Подключение по локальной сети — свойства»**, закладка **«Общие»** (рис. Л.5.6). Тут у полі **«Подключение через:»** зазначається тип мережного адаптера, за допомогою якого буде здійснюватися підключення до комп'ютерної мережі. У полі **«Отмеченные компоненты используются этим подключением:»** зазначаються мережні компоненти (клієнти, служби, протоколи), які необхідні для роботи комп'ютера в мережі:

— **«Клиент для сетей *Microsoft*»**;
— **«Служба доступа к файлам и принтерам сетей *Microsoft*»**;
— **«Протокол Интернета (TCP/IP)»**.

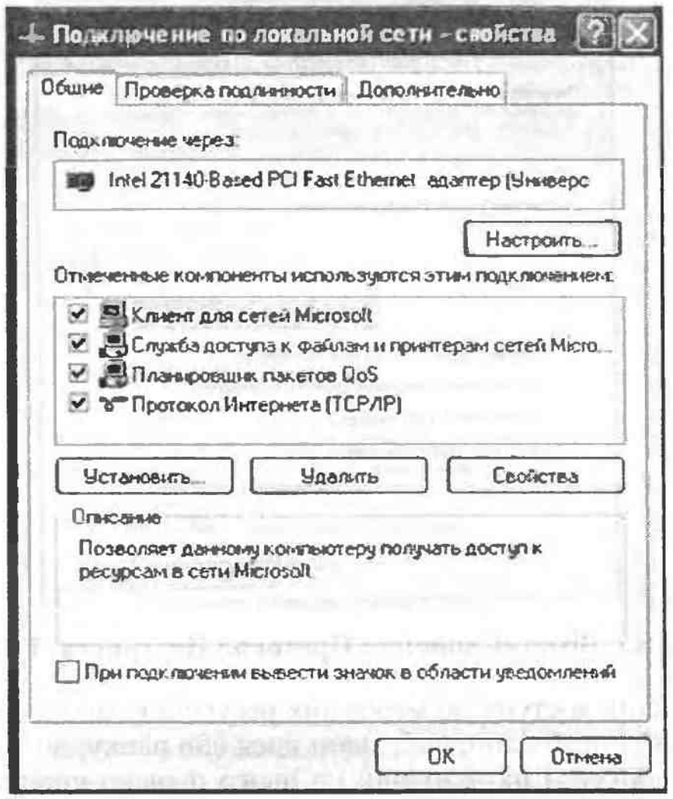


Рис. Л.5.6. Закладка **«Общие»**

Ці мережні компоненти встановлені за замовчуванням у ході інсталяції операційної системи *Windows* XP. Для додавання нового компонента потрібно лівою кнопкою миші натиснути пункт **«Установить...»** та у вікні, що з'явиться, вибрати новий мережний компонент (клієнт, службу, протокол), після чого натиснути пункт **«Добавить...»** і, вибравши необхідний компонент, натиснути **«ОК»**. Для зміни властивостей мережного компонента треба виділити його лівою кнопкою миші та натиснути пункт меню **«Свойства»** (там, де він доступний) і після внесення необхідних змін натиснути **«ОК»**. Наприклад, якщо виділити компонент **«Протокол Интернета (TCP/IP)»** та натиснути пункт меню **«Свойства»**, з'явиться вікно **«Свойства: Протокол Интернета (TCP/IP)»** (рис. Л.5.7), де можна задати автоматичне одержання IP-адреси або задати її та маску підмережі вручну, відтак натиснути **«ОК»**.

Для того щоб зміни набули чинності після внесення змін у мережні настроювання у вікні **«Подключение по локальной сети - свойства»** натиснути **«ОК»**.

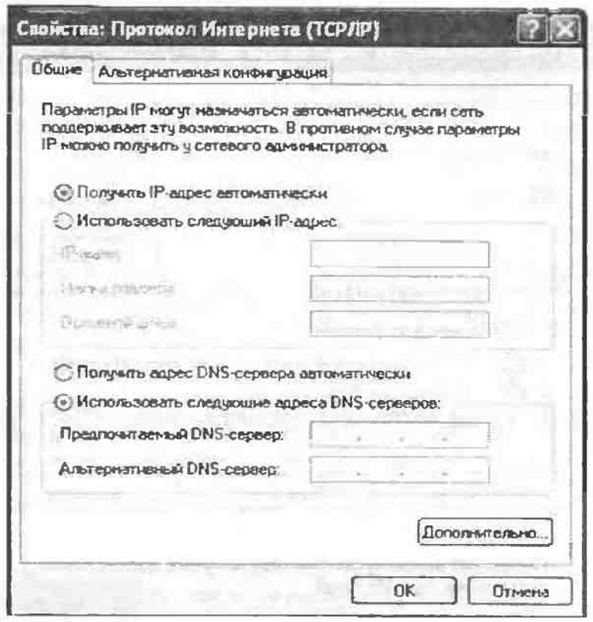


Рис. Л.5.7. Вікно **«Свойства: Протокол Интернета (TCP/IP)»**

Для надання доступу до мережних ресурсів комп'ютера (найпростіший варіант) необхідно, вибравши диск або папку, до якого потрібно надати доступ, і натиснувши на нього правою кнопкою миші, у меню, що з`явилося, вибрати пункт **«Общий доступ и безопасность...»**. У вікні, що з'явиться, — **«Свойства: (имя папки)»**, вибрати закладку **«Доступ»** та в полі **«Сетевой совместный доступ и безопасность»** правою кнопкою миші натиснути на пункті меню **«Если вы понимаете потенциальную опасность, но все равно хотите включить общий доступ без помощи мастера, щелкните здесь»**. Закладка **«Доступ»** змінить свій вигляд (рис. Л.5.8).

Необхідно відзначити пункт **«Открыть общий доступ к этой папке»**, а в полі **«Общий ресурс»** вказати ім'я мережного ресурсу (за замовчуванням — власне ім'я папки), після чого натиснути **«Применить»** та **«ОК»**.

Створений мережний ресурс буде мати права доступу «тільки читання». Якщо потрібно надати право на запис та видалення файлів, що перебувають на цьому мережному ресурсі, у вікні (див. рис. Л.5.8) слід відзначити пункт **«Разрешить изменение файлов по сети»**. З міркувань безпеки не рекомендується надавати загальний доступ до всього диска. Цей спосіб створення мережного ресурсу та надання до нього доступу є єдино можливим в операційній системі *Windows* XP *Home Edition*. В *Windows* XP *Professional* існують можливості задавати різні паролі доступу до мережних ресурсів для різних груп користувачів, що мають різні права доступу.

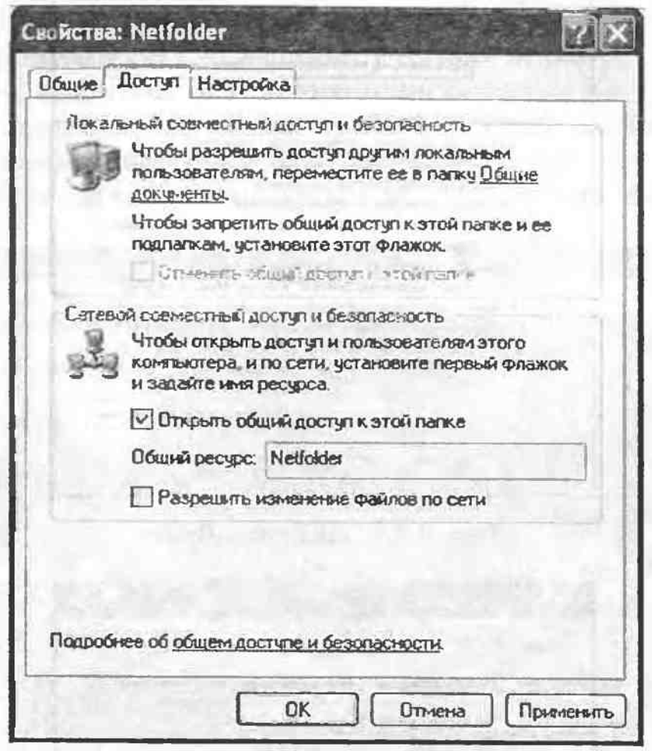


Рис. Л.5.8. Закладка **«Доступ»**

Для того, щоб ця можливість стала доступною, потрібно подвійним натисканням лівої кнопки миші звернутися до ярлика **«Мой компьютер»**, який розташований на робочому столі. З'явиться вікно **«Мой компьютер»**, в якому на панелі інструментів необхідно вибрати пункт **«Сервис»** та у меню, що з'явиться, вибрати пункт **«Свойства папки...»** та перейти на закладку **«Вид»** (рис. Л.5.9). У полі **«Дополнительные параметры:»** слід прибрати позначку навпроти пункту **«Использовать простой общий доступ к файлам (рекомендуется)»**, після чого треба натиснути **«Применить»** й **«ОК»**. Тепер вигляд закладки **«Доступ»** у властивостях об'єкта змінить свій вигляд (рис. Л.5.10), надаючи можливість робити доступними мережні ресурси певним користувачам або групам користувачів з використанням паролів доступу.

Для ефективної роботи з мережними ресурсами необхідно створити облікові записи користувачів та груп користувачів.

Рис. Л.5.9. Закладка «**Вид**»

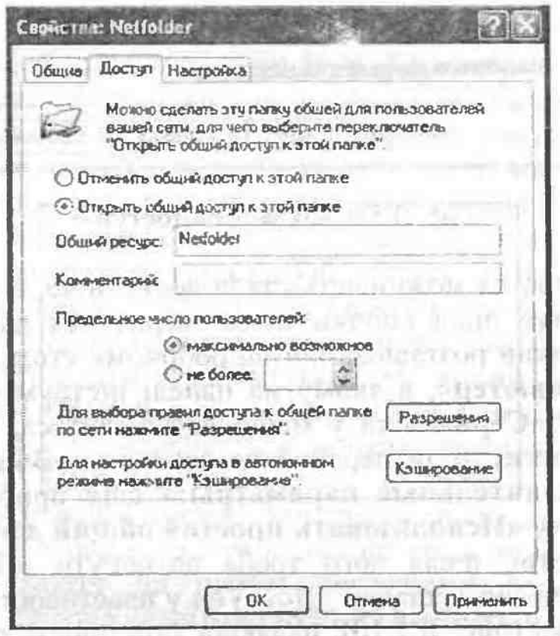


Рис. Л.5.10. Закладка «**Доступ**»

Для створення нового облікового запису користувача потрібно після натискання кнопки «**Пуск**» у меню вибрати пункт «**Настройка**», далі пункт «**Панель управления**». У вікні «**Панель управления**» необхідно вибрати пункт «**Учетные записи пользователей**» та виконати подвійне натискання лівої кнопки миші на ньому. З'явиться вікно «**Учетные записи пользователей**» (рис. Л.5.11), у якому потрібно вибрати пункт «**Создание учетной записи**», далі у відповідному полі вводиться ім'я нового користувача та після натиснення «**Далее >**» задається тип облікового запису користувача: «**Администратор компьютера**» або «**Ограниченная запись**» (звичайний користувач), після чого натискається «**Создать учетную запись**».

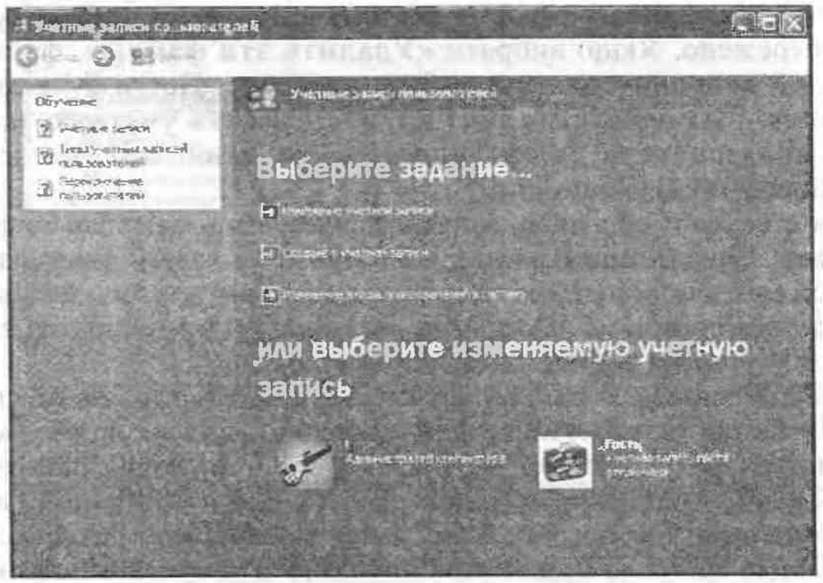


Рис. Л.5.11. Вікно «**Учетные записи пользователей**»

У вікні «**Учетные записи пользователей**» у списку облікових записів користувачів додається новий користувач. При натисканні на цьому обліковому записі лівою кнопкою миші з'являється можливість його відредагувати. У редагуванні облікового запису користувача є можливості:

- «**Изменение имени**»;
- «**Создание пароля**»;
- «**Изменение изображения**»;
- «**Изменение типа учетной записи**»;
- «**Удаление учетной записи**».

При виборі пункту «**Изменение имени**» з'явиться вікно, у відповідному полі якого потрібно ввести нове ім'я користувача та натиснути «**Сменить имя**».

При виборі пункту **«Создание пароля»** з'явиться вікно, у відповідному полі якого небохідно ввести новий пароль та пароль для підтвердження, після чого натиснути **«Создать пароль»**.

При виборі пункту **«Изменение типа учетной записи»** потрібно вибрати для користувача тип облікового запису **«Администратор компьютера»** або **«Ограниченная запись»** (звичайний користувач) і натиснути **«Изменить тип учетной записи»**.

При виборі пункту **«Удаление учетной записи»** з'явиться вікно, у якому буде задане запитання **«Хотите сохранить файлы, принадлежавшие "имя пользователя"?»**. Якщо вибрати **«Сохранить эти файлы»**, файли та домашній каталог користувача буде збережено. Якщо вибрати **«Удалить эти файлы»**, файли та домашній каталог користувача буде видалено. Потім буде задане запитання **«Вы действительно хотите удалить учетную запись "имя пользователя"?»**, у відповідь на який слід натиснути **«Удалить учетную запись»**.

При створенні облікових записів користувачів потрібно створити обліковий запис Адміністратора комп'ютера та задати йому пароль. Особисті дані адміністратора комп'ютера міститимуться в домашньому каталозі користувача: *C:\Documents and Settings*\Адміністратор та будуть недоступними для інших користувачів.

Облікові записи користувачів можна створювати іншим способом. Необхідно натиснути правою кнопкою миші на ярлику **«Мой компьютер»**, розташованому на робочому столі, та у меню, що з'явиться, вибирати пункт **«Управление»**. Відтак з'явиться вікно **«Управление компьютером»** (рис. Л.5.12), що складається з двох полів, у якому лівою кнопкою миші потрібно відзначити пункт **«Локальные пользователи и группы»**. У полі вікна з'являться дві папки: **«Пользователи»** та **«Группы»**.

Далі необхідно перейти в папку **«Пользователи»**, у якій зберігаються облікові записи користувачів (рис. Л.5.13). Для додавання нового користувача потрібно правою кнопкою миші натиснути на вільному місці в полі, де відображається список облікових записів користувачів.

З'явиться меню, у якому слід вибрати пункт **«Новый пользователь...»**. Облікові записи **«Администратор»** й **«Гость»** є вбудованими обліковими записами в системі. Обліковий запис **«Гость»** рекомендується відключити з міркувань безпеки. З'явиться вікно **«Новый пользователь»** (рис. Л.5.14), у відповідних полях якого вказується ім'я користувача та пароль, якщо це необхідно, потім натискається **«Создать»** та **«Закрыть»**. Для редагування облікового запису користувача необхідно в списку облікових записів вибрати

потрібний та натиснути на ньому правою кнопкою миші. З'явиться меню, за допомогою відповідних пунктів якого обліковий запис можливо перейменувати, видалити, задати йому пароль. Якщо вибрати пункт меню **«Свойства»**, з'явиться вікно (рис. Л.5.15).

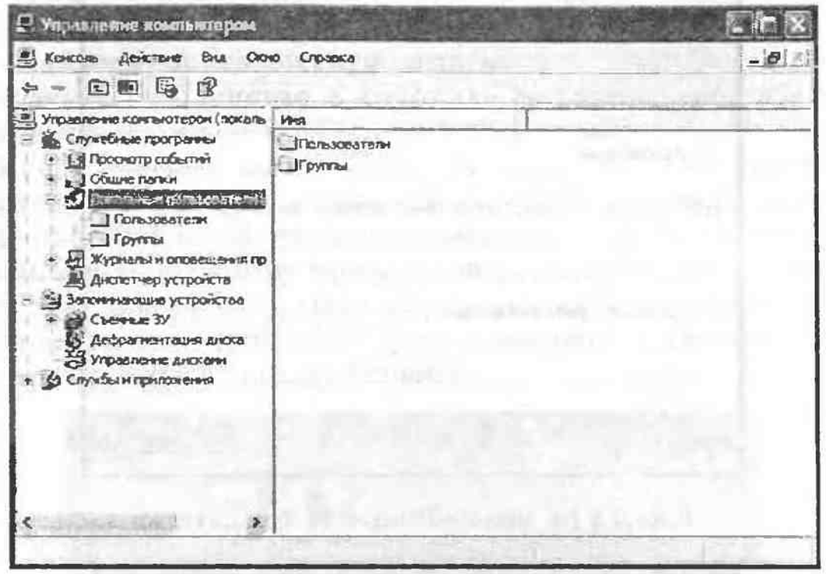


Рис. Л.5.12. Вікно **«Управление компьютером»**

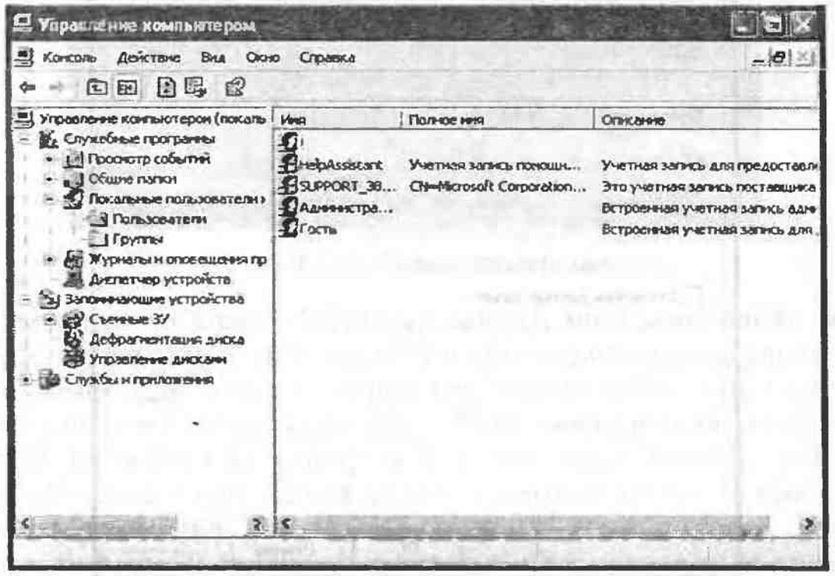


Рис. Л.5.13. Папка **«Пользователи»**

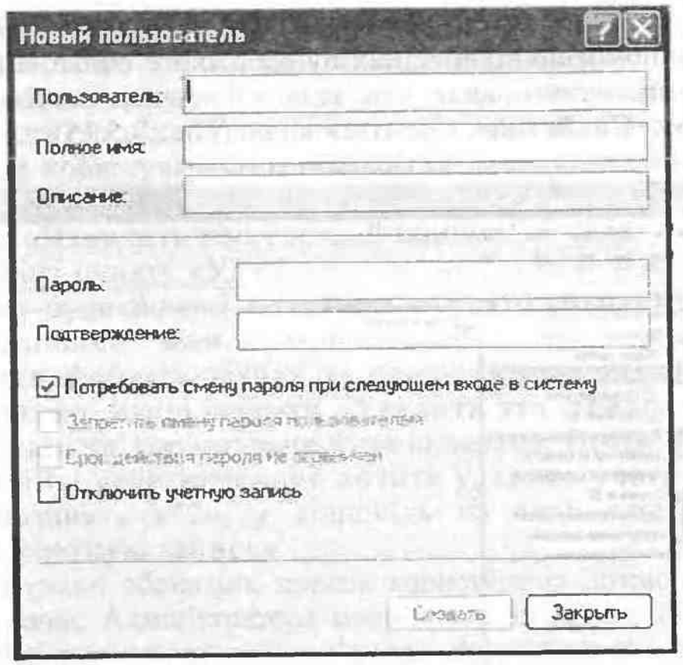


Рис. Л.5.14. Вікно **«Новый пользователь»**

Рис. Л.5.15. Вікно **«Свойства»**

У вікні **«Свойства»** на закладці **«Общие»** доступні такі настроювання:

— **«Потребовать смену пароля при следующем входе в систему»;**
— **«Запретить смену пароля пользователем»;**
— **«Срок действия пароля не ограничен»;**
— **«Отключить учетную запись»;**
— **«Заблокировать учетную запись».**

На закладці **«Членство в группах»** зазначені група або групи користувачів, до складу яких входить користувач.

У вікні **«Учетные записи пользователей»** за допомогою пункту **«Изменение входа пользователей в систему»** можна змінювати зовнішній вигляд вікна привітання системи. Якщо відзначено пункт **«Использовать страницу приветствия»**, то вікно привітання буде мати вигляд (рис. Л.5.16), якщо забрати цю оцінку, вікно вітання матиме інший вигляд (рис. Л.5.17) — класичний. Класичний вигляд привітання кращий з погляду безпеки.

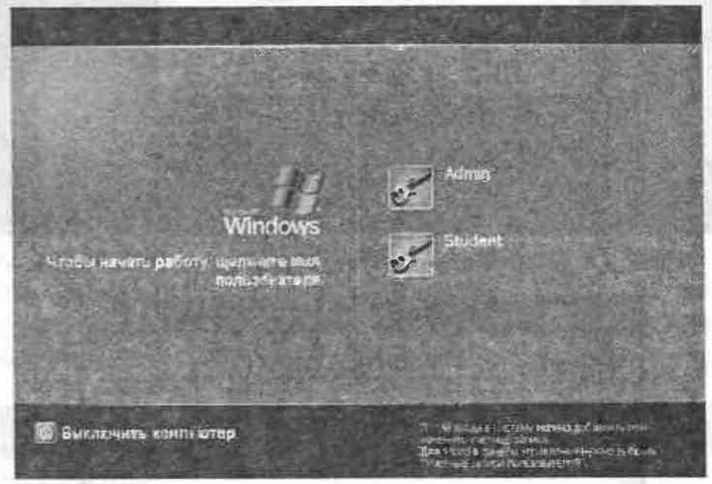


Рис. Л.5.16. Вікно привітання

За допомогою вікна **«Учетные записи пользователей»** можна створювати тільки облікові записи нових користувачів, але не груп користувачів. Для створення груп користувачів потрібно натиснути правою кнопкою миші на ярлику **«Мой компьютер»**, який розташований на робочому столі, та у меню, що з'явиться, вибирати пункт **«Управление»**. Після цього з'явиться вікно **«Управление компьютером»** (див. рис. Л.5.12), у якому лівою кнопкою миші необхідно відзначити пункт **«Локальные пользователи и группы»**. Заходимо в папку **«Группы»**, у якій перебуває список груп користувачів. Для створення нової групи користувачів потрібно правою кнопкою

миші натиснути на вільному місці в полі, де відображається список груп користувачів. З'явиться меню, у якому треба вибрати пункт «**Создать группу...**». З'явиться вікно «**Новая группа**» (рис. Л.5.18), у відповідних полях якого задається ім'я групи користувачів і натискається «**Создать**» та «**Закрыть**». Для додавання в групу нових користувачів потрібно натиснути правою кнопкою миші на групі користувачів та у меню, що з'явиться, вибрати пункти «**Добавить в группу...**» або «**Свойства**». З'явиться вікно «**Свойства**» (рис. Л.5.19), у полі «**Члены группы**» якого відображається список користувачів, зареєстрованих у цій групі.

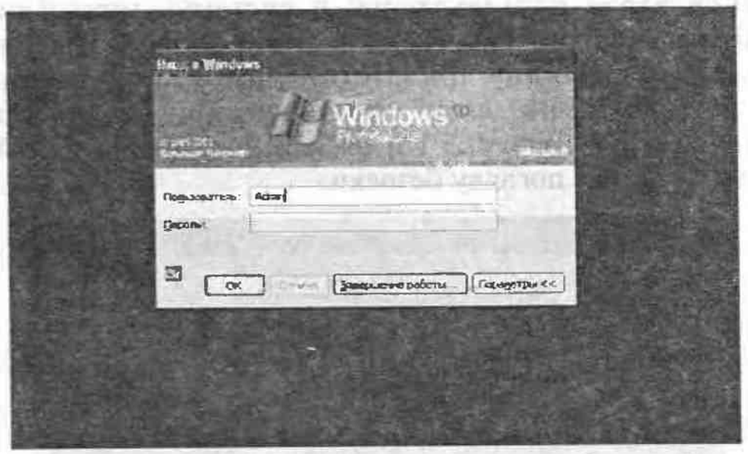


Рис. Л.5.17. Вікно привітання (класичний вигляд)

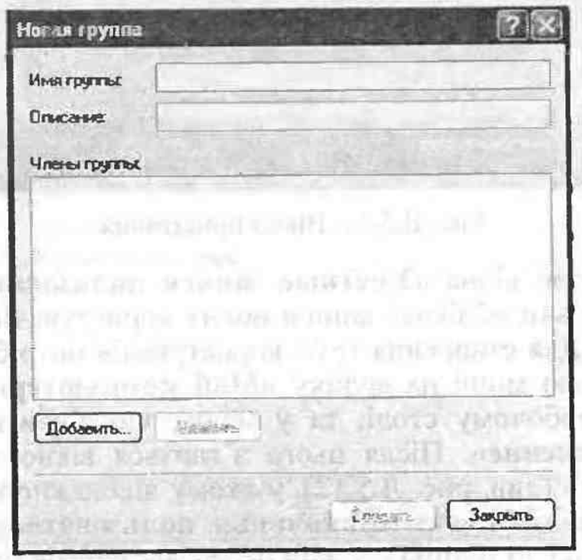


Рис. Л.5.18. Вікно «**Новая группа**»

Рис. Л.5.19. Вікно «**Свойства**»

Для додавання нового користувача потрібно натиснути «**Добавить...**», після чого з'явиться вікно «**Выбор**» (рис. Л.5.20), у якому треба натиснути «**Дополнительно...**» та у вікні, що з'явиться (рис. Л.5.21), у якому здійснюється вибір користувачів для додавання в групу, натиснути «**Поиск**».

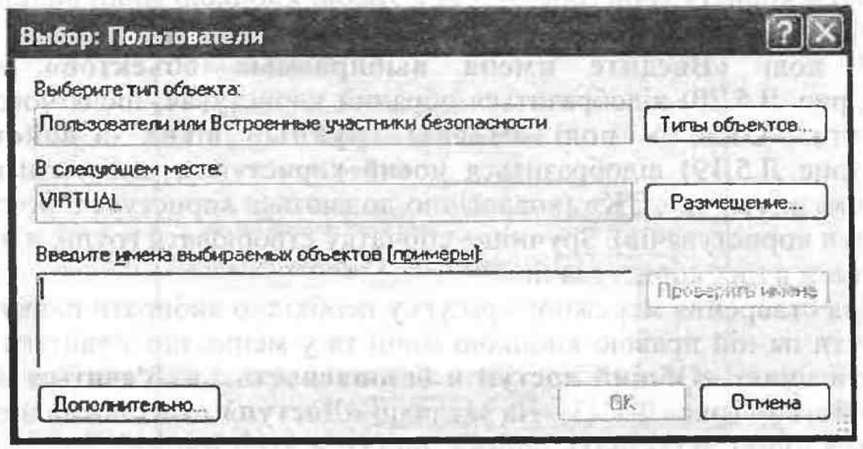


Рис. Л.5.20. Вікно «**Выбор**»

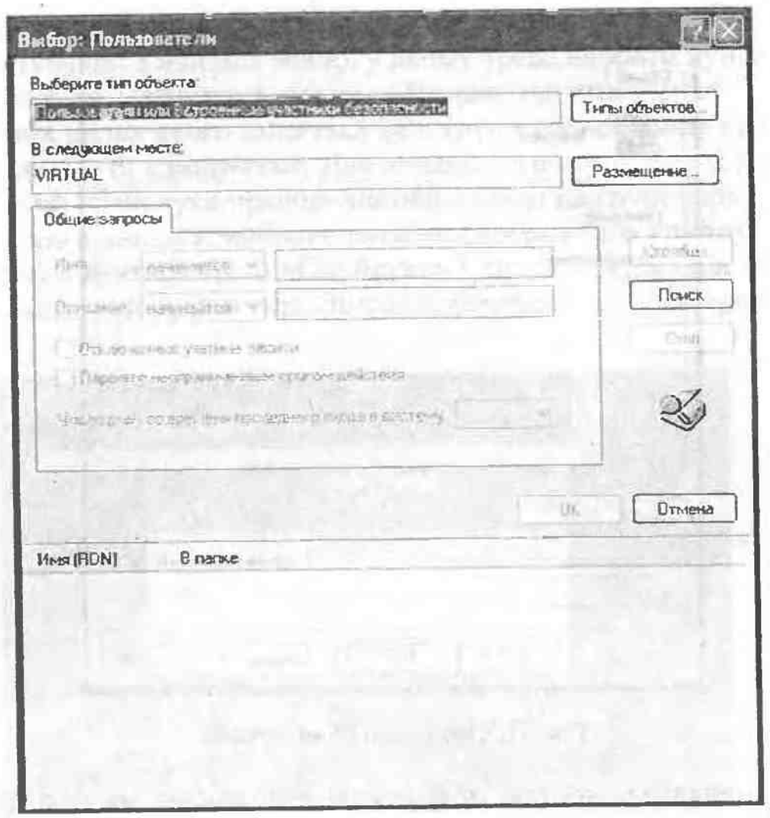


Рис. Л.5.21. Вікно **«Выбор», «Дополнительно...»**

Відтак в полі вікна відображаються всі зареєстровані користувачі та групи користувачів (рис. Л.5.22). Лівою кнопкою миші виділяємо потрібного користувача й натискаємо **«ОК»**.

У полі **«Введите имена выбираемых объектов»** вікна (див. рис. Л.5.20) відобразиться обраний користувач, після чого натиснути **«ОК»**. У полі **«Члены группы»** вікна **«Свойства»** (див. рис. Л.5.19) відобразиться новий користувач, далі натиснути **«Применить»** та **«ОК»** (аналогічно додаються користувачі до складу груп користувачів). Зручніше спочатку створювати групи, а потім додавати в них користувачів.

Для створення мережного ресурсу необхідно вибирати папку, натиснути на ній правою кнопкою миші та у меню, що з'явиться, вибирати пункт **«Общий доступ и безопасность...»**. З'явиться вікно **«Свойства»** (рис. Л.5.23). На закладці **«Доступ»** цього вікна відзначається пункт **«Открыть общий доступ к этой папке»**, а потім натискається **«Разрешения»**.

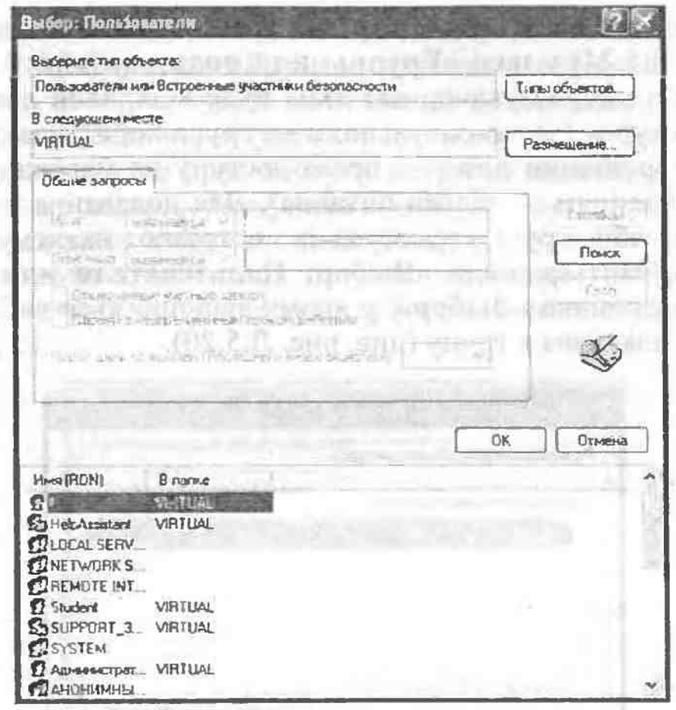


Рис. Л.5.22. Вікно **«Выбор», «Поиск»**

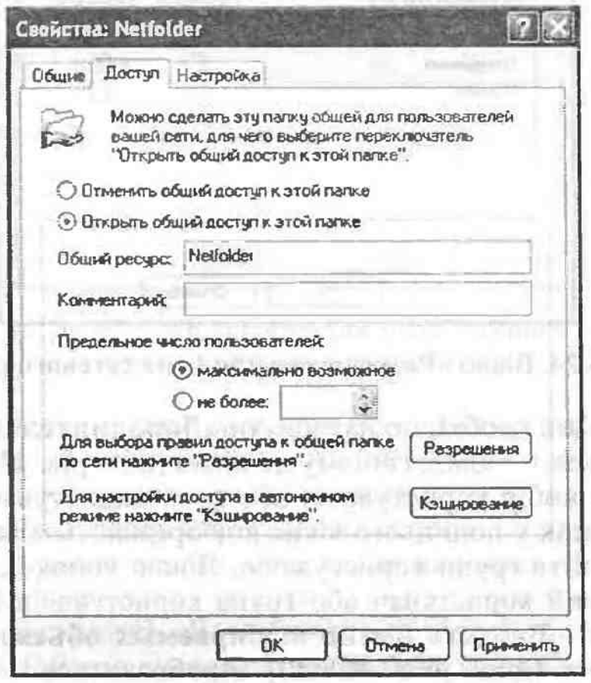


Рис. Л.5.23. Вікно **«Свойства»**

У вікні, що з'явиться, «**Разрешения для (имя сетевого ресурса)**» (рис. Л.5.24) у полі «**Группы или пользователи:**» зазначений список груп і користувачів, для яких буде відкритий доступ до мережного ресурсу (за замовчуванням — група користувачів «**Все**»), а в полі «**Разрешения для**» — права доступу до мережного ресурсу (за замовчуванням — тільки читання). Для додавання в список користувачів або груп користувачів потрібно натиснути «**Добавить...**». З'явиться вікно «**Выбор: Пользователи или Группы**», аналогічне до вікна «**Выбор**», у якому здійснюється вибір користувачів для додавання в групу (див. рис. Л.5.20).

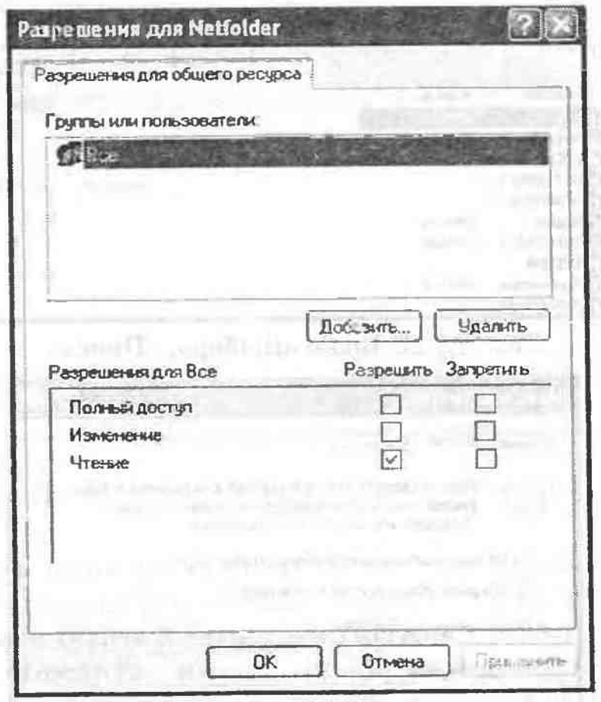


Рис. Л.5.24. Вікно «**Разрешения для (имя сетевого ресурса)**»

В цьому вікні необхідно натиснути «**Дополнительно...**» та у вікні, що з'явиться, — аналогічному до вікна (див. рис. Л.5.21), у якому здійснюється вибір користувачів або груп користувачів, натиснути «**Поиск**». Відтак у полі цього вікна відображаються всі зареєстровані користувачі та групи користувачів. Лівою кнопкою миші виділяється потрібний користувач або група користувачів і натискається «**ОК**». У полі «**Введите имена выбираемых объектов**», аналогічного до вікна (див. рис. Л.5.20) відобразиться обраний користувач або група користувачів, після чого натискається «**ОК**».

У вікні, що з'явиться, «**Разрешения для (имя сетевого ресурса)**» (див. рис. Л.5.24) у полі «**Группы или пользователи:**» у списку груп і користувачів, які будуть мати доступ до мережного ресурсу, додається обраний нами користувач або група користувачів (рис. Л.5.25). Для редагування його прав доступу потрібно правою кнопкою миші виділити в списку користувача або групу користувачів, а в полі «**Разрешения для**» лівою кнопкою миші ставити або видаляти позначки навпроти відповідних пунктів «**Чтение**», «**Изменение**» й «**Полный доступ**», що перебувають у стовпчиках «**Разрешить**» та «**Запретить**».

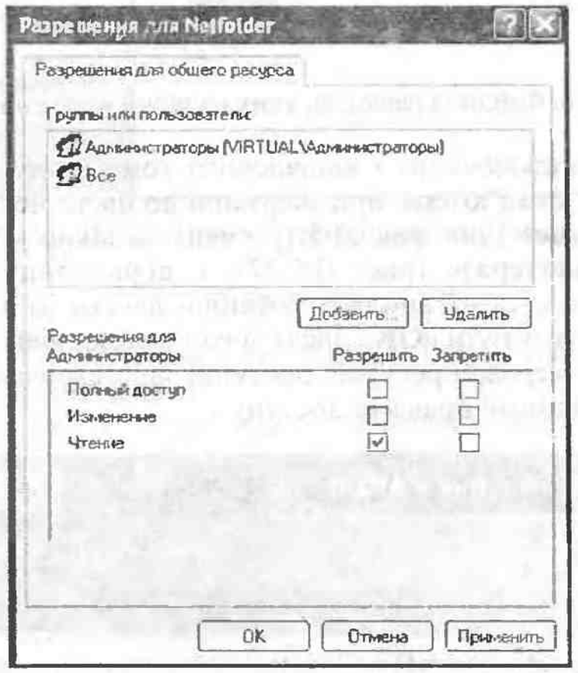


Рис. Л.5.25. Вікно «**Разрешения для (имя сетевого ресурса)**» з новою групою користувачів

Якщо поставити позначку в стовпчику «**Запретить**», то всі позначки в стовпчику «**Разрешить**» будуть зняті автоматично і доступ відповідного користувача або групи користувачів до цього ресурсу буде заборонений. Якщо потрібно видалити користувача або групу користувачів із цього списку, слід лівою кнопкою миші виділити в списку користувача або групу користувачів і натиснути «**Удалить**». Так, групу користувачів «**Все**» бажано видалити зі списку з міркувань безпеки. Ці настроювання повинні бути виконані для створення мережного ресурсу на жорсткому диску, що має файлову систему

FAT32. Якщо на жорсткому диску файлова система NTFS, то у вікні **«Свойства»** (див. рис. Л.5.23), крім закладки **«Доступ»**, де здійснюються описані вище настроювання, присутня закладка **«Безопасность»**, де повинні бути продубльовані ті самі настроювання, що й на закладці **«Доступ»**. Для того, щоб зміни набули чинності, натискається **«Применить»** та **«ОК»** у цьому вікні, а потім у вікні **«Свойства»** (див. рис. Л.5.23). Файли та папки, до яких наданий мережний доступ, мають такий вигляд (рис. Л.5.26).

Рис. Л.5.26. Файли та папки, до яких наданий мережний доступ

Тепер при підключенні з віддаленого комп'ютера до мережних ресурсів цього комп'ютера, при звертанні до нього через ярлик **«Сетевое окружение»** (див. рис. Л.5.1) з'явиться вікно **«Подключение к (имя компьютера)»** (рис. Л.5.27), у відповідних полях якого **«Пользователь:»** та **«Пароль:»** потрібно ввести ім'я користувача, його пароль та натиснути **«ОК»**, після чого з'явиться вікно (рис. Л.5.28), що відображає мережні ресурси, доступні зареєстрованому користувачеві з відповідними правами доступу.

Рис. Л.5.27. Вікно «Подключение к (имя компьютера)»

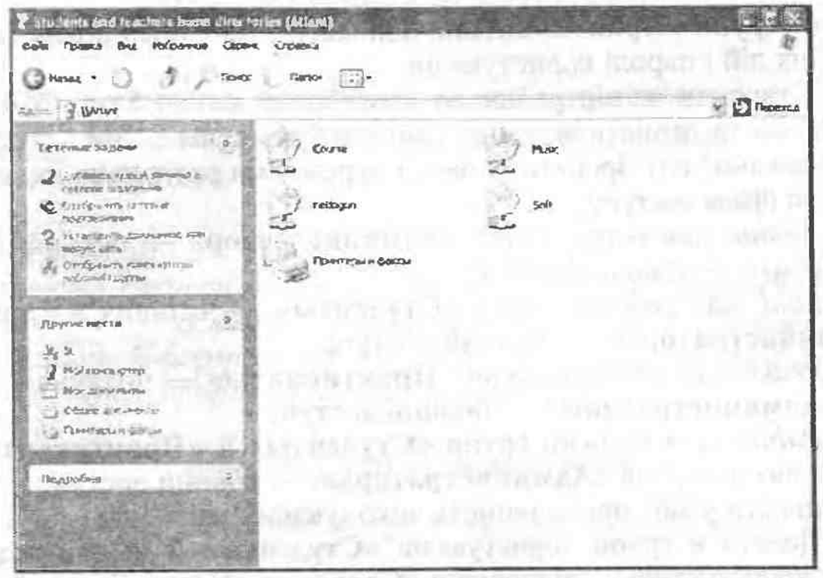


Рис. Л.5.28. Вікно з мережними ресурсами

Завдання

1. На віртуальному комп'ютері запустити віртуальну операційну систему *Windows* XP.

2. Задати ім'я комп'ютера, що не повторюється в цій робочій групі, та назву робочої групи ІІТ або тієї, котру вкаже викладач. В описі комп'ютера вказати VIRTUAL. Записати цю інформацію у звіт.

3. Перевірити, які мережні компоненти встановлені в системі, й записати їх у звіт.

4. Створити робочі групи **«Студенты»** та **«Практиканты»**. Записати у звіт послідовність виконуваних дій.

5. Створити користувачів: «*Boss*», **«Петя»**, **«Костя»**, **«Света»**, **«Настя»**. Користувача «*Boss*» внести в робочу групу **«Администраторы»** та задати йому й користувачеві «Адміністратор» пароль завдовжки не менш ніж шість символів. Усі користувачі повинні мати пароль — всі користувачі однієї групи можуть мати один пароль. У вікні **«Свойства»** всіх користувачів поставити позначки навпроти **«Запретить смену пароля пользователем»** і **«Срок действия пароля не ограничен»**, інші позначки прибрати. Записати у звіт послідовність виконуваних дій та паролі користувачів.

6. Додати в групу користувачів **«Студенты»** користувачів **«Петя»** та **«Костя»**, а в групу користувачів **«Практиканты»** — користувачів **«Света»** та **«Настя»**. Установити паролі для користувачів — один для всіх користувачів групи **«Студенты»** й один для всіх корис-

тувачів групи «**Практиканты**». Записати у звіт послідовність виконуваних дій і паролі користувачів.

7. Створити на віртуальному комп'ютері папки *Stud*, *Prakt*, *Public*, *Admin* та помістити в них файли з будь-яким ім'ям і розширенням (бажано*.*txt*). Зробити ці папки мережними ресурсами, надавши до них такі права доступу:

– *Admin*: для користувача «**Администратор**» — повний доступ, для «*Boss*» — тільки читання;

– *Stud*: для робочої групи «**Студенты**» — читання й запис, для «**Администраторы**» — повний доступ;

– *Prakt*: для робочої групи «**Практиканты**» — читання й запис, для «**Администраторы**» — повний доступ;

– *Public*: для робочої групи «**Студенты**» й «**Практиканты**» — тільки читання, для «**Администраторы**» — повний доступ.

Записати у звіт послідовність виконуваних дій.

8. Додати в групи користувачів «**Студенты**» й «**Практиканты**» нових користувачів — відповідно «**Слава**» та «**Катя**». Задати їм паролі. Записати у звіт послідовність виконуваних дій і паролі користувачів.

9. Видалити облікові записи користувачів «Костя» та «Настя». Записати у звіт послідовність виконуваних дій.

10. Змінити зовнішній вигляд вікна вітання системи на класичний. Записати у звіт послідовність виконуваних дій.

11. Увійти з реального комп'ютера на віртуальний як Адміністратор та звернутися до мережних ресурсів, потім з одного віртуального комп'ютера на інший як звичайний користувач кожної з груп. Перевірити, які мережні ресурси доступні та з якими правами доступу. Порівняти результати з настроюваннями з п. 7. Якщо при звертанні до віртуального комп'ютера з реального папка «**Мережне оточення**» реального комп'ютера перестане реагувати на команди, її необхідно згорнути (не закривати!) і повторити спробу входу. Результати записати до звіту.

12. Зробити копіювання файлів з реального комп'ютера на віртуальний і з віртуального на реальний. Результати записати до звіту.

Питання для самоперевірки

1. Призначення мережного компонента «**Служба**».
2. Призначення мережного компонента «**Клиент**».
3. Охарактеризувати режим «**Полный доступ**» для мережного ресурсу.
4. Для чого призначений мережний компонент «**Протокол**»?
5. Особливості режиму «**Определяется паролем**» для мережного ресурсу.

ЛАБОРАТОРНА РОБОТА № 6

Вивчення програмного забезпечення для роботи в локальних комп'ютерних мережах

Мета роботи:

вивчення спеціалізованого програмного забезпечення для роботи в локальних комп'ютерних мережах, яке використовується для збору інформації про мережні настроювання комп'ютерів та пошуку інформації на мережних ресурсах.

Короткі теоретичні відомості

Ім'я хосту — ім'я комп'ютера в мережі, відображуване замість його IP-адреси.

MAC-адреса — апаратна адреса мережної карти, задається при її виготовленні.

IP-адреса — ідентифікатор комп'ютера у комп'ютерній мережі, що працює за протоколом TCP/IP. Задається як 32-бітне число, що вказується в байтах і має чотири числові сегменти, розділені крапками. Одна частина цієї адреси являє собою адресу мережі, а інша використається для позначення конкретного комп'ютера в цій комп'ютерній мережі. Адреса мережі позначає мережу, частиною якого є даний комп'ютер. Мережна частина IP-адреси займає перші три сегменти, а адреса машини — останній сегмент. Наприклад, в IP-адресі 196.162.0.119 мережною частиною є цифри 196.162.0, а адресою комп'ютера в комп'ютерній мережі 119. IP-адреса не повинна повторюватися в локальній мережі.

Маска підмережі використається для одержання адреси комп'ютерної мережі. Являє собою чотири числові сегменти, розділені крапками. При визначенні маски підмережі IP-адреса комп'ютера виступає в ролі трафарету. Наприклад, маска підмережі для IP-адреси 196.162.0.119 буде 255.255.255.0. У цьому випадку мережна частина, 196.162.0, замінена на 255.255.255, а машинна частина, 119, замінена на 0.

Для створення локальної комп'ютерної мережі в приватному адресному просторі заброньовано три блоки IP-адрес:

— блок A: 10.0.0.0 – 10.255.255.255;
— блок B: 172.16.0.0 – 172.31.255.255;
— блок C: 192.168.0.0 – 192.168.255.255.

Розрізняють динамічну й статичну IP-адреси.

Статична IP-адреса присвоюється вручну в мережних настроюваннях комп'ютера. Динамічну IP-адресу комп'ютер у локальній мережі одержує автоматично після його включення.

Програма TCP *Net View*

Програма TCP *Net View* проста у використанні і не вимагає інсталяції на комп'ютер. Інтерфейс програми наведено на рис. Л.6.1. Програма TCP *Net View* використається для відображення інформації про комп'ютери в мережі, а саме: ім'я мережного ресурсу, ім'я хосту, IP і MAC-адреси і додаткову інформацію — коментар до ресурсу.

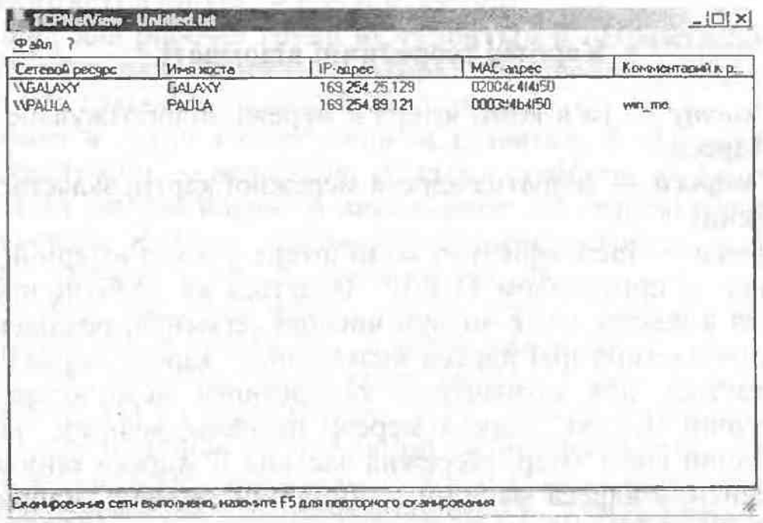


Рис. Л.6.1. Інтерфейс програми TCP *Net View*

Використання цієї програми дає змогу створювати файли звіту у форматі **.txt*. Для створення файла звіту необхідно у команді **«Файл»** вибрати **«Сохранить как»** (**«Сохранить»**) і вказати місце, де буде збережено файл звіту.

Програма має російський та англійський інтерфейси. Вибір мови здійснюється за допомогою команди **«Файл»** ⇒ *Switch to English* (*Russian*). Для відновлення інформації у вікні звіту потрібно вибрати **«Файл»** ⇒ **«Обновить»** або натиснути клавішу F5.

Програма LAN *Search*

Програма LAN *Search* призначена для пошуку файлів у локальній мережі на доступних мережних ресурсах. Інтерфейс програми представлено на рис. Л.6.2.

Пошук виконується на всіх комп'ютерах, у всіх робочих групах або доменах мережі. Для пошуку необхідного файла потрібно ввести у вікні пошуку ім'я шуканого файла або маску пошуку відповідно до синтаксису MS-DOS. Наприклад, для пошуку файлів з розширенням «**.mp3*» потрібно просто ввести **.mp3* у вікно пошуку і за допомогою миші натиснути кнопку **«Найти»**. Після завершення пошуку користувач має можливість копіювати на свій комп'ютер усі необхідні дані. Для виконання цієї операції створюється папка (каталог для файлів, що зберігаються, C:\ *ім'я папки*). При копіюванні файлів виділити необхідну групу файлів або один файл і натиснути кнопку **«Скачать»**.

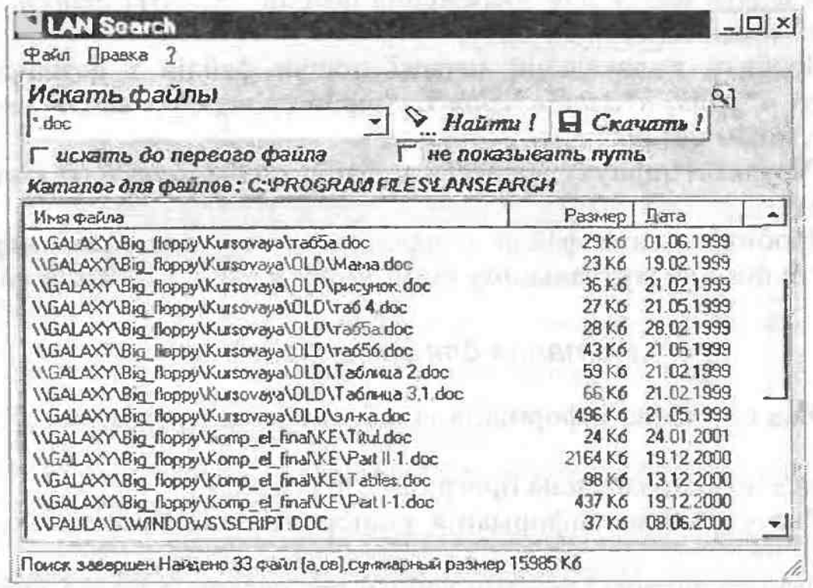


Рис. Л.6.2. Інтерфейс програми LAN *Search*

Для збереження результатів пошуку інформації необхідно зберегти результати пошуку за допомогою команди **«Файл»** ⇒ **«Сохранить»** або натиснувши клавішу F2.

При копіюванні файлів можна змінити каталог для збереження, виконавши команди **«Файл»** ⇒ **«Измепить каталог»**.

Програма має додаткові можливості:

— пошук може виконуватися до першого файла, що відповідає умовам пошуку;

— шлях до файла можна не показувати.

Завдання

1. Запустити програму TCP *Net View* на віртуальному комп'ютері. Переключити мову інтерфейсу на англійську мову, а потім на російську.

2. Створити і зберегти файл звіту з назвою *otchet.txt*.

3. В отриманому звіті знайти інформацію про віртуальні комп'ютери і записати її в звіт лабораторної роботи. Віртуальні комп'ютери зазначені в коментарях до ресурсу як VIRTUAL.

4. Інсталювати програму LAN *Search* по локальній мережі на віртуальний комп'ютер. Для цього, використовуючи ярлик «Мережне оточення», необхідно зайти на реальний комп'ютер (попередньо з'ясувавши його мережне ім'я), у папку C:\ *Install* і запустити файл *lansearch.exe*. Після інсталяції при першому запуску програми потрібно задати папку для збереження файлів – C:\ *Net Search*. Порядок інсталяції занести до звіту.

5. Зробити в локальній мережі пошук файлів з розширенням «*.doc», «*.jpg», «*.txt», «*.zip» і «*mp3» спочатку з відображенням шляху, потім без нього.

6. Результат пошуку зберегти у файлі *otchet_search.txt* і занести до звіту.

7. Зробити пошук файла з заданим ім'ям у локальній мережі і зберегти його на віртуальному комп'ютері в папці C:*Netsearch*.

Питання для самоперевірки

1. Яка службова інформація відображається програмою TCP *Net View*?
2. Для чого призначена програма LAN *Search*?
3. Яка службова інформація відображається програмою LAN *Search*?
4. Які можливості і режими роботи має програма LAN *Search*?
5. У чому різниця між поняттями IP та MAC-адреса?
6. У чому різниця між динамічною та статичною IP-адресою?

ЛАБОРАТОРНА РОБОТА № 7

Вивчення програми для обміну текстовими повідомленнями у локальній мережі *VYPRESS CHAT*

Мета роботи:

вивчення спеціалізованого програмного забезпечення для роботи в локальних комп'ютерних мережах, призначеного для обміну текстовими повідомленнями між користувачами.

Короткі теоретичні відомості

Програма *Vypress Chat* призначена для обміну текстовими повідомленнями в реальному часі між користувачами локальної комп'ютерної мережі. Інтерфейс програми представлено на рис. Л.7.1.

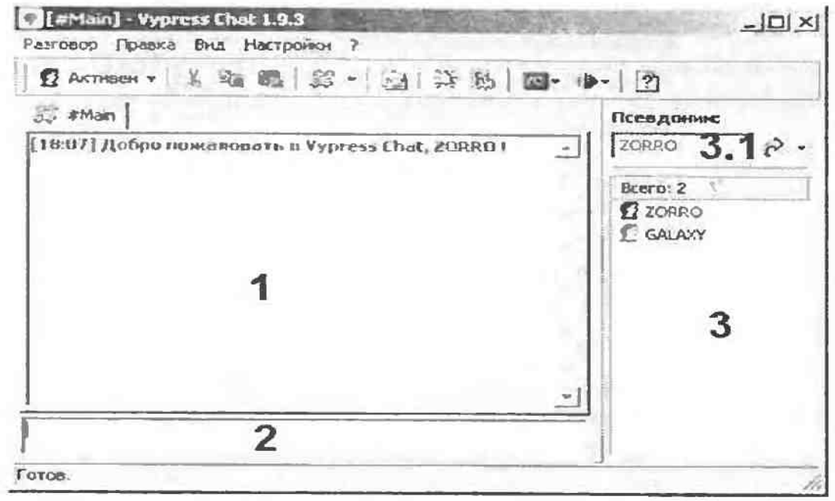


Рис. Л.7.1. Інтерфейс програми *Vypress Chat*

Інтерфейс програми має такі елементи:

– «**1**» – поле, в якому відображається зміст каналу обміну текстовими повідомленнями;

– «**2**» – поле, в яке вводиться текст повідомлення для відправлення іншим користувачам;

– «**3**» – поле, в якому відображається список зареєстрованих користувачів і режим їхньої роботи;

– «**3.1**» – поле, в якому відображається псевдонім користувача.

Для зміни свого псевдоніму необхідно ввести в полі «**3.1**» новий псевдонім і натиснути клавішу ENTER.

Після запуску програми за замовчуванням створюється загальний канал обміну текстовими повідомленнями між усіма зареєстрованими користувачами – # *Main*. Крім загального каналу обміну повідомленнями, є можливість створення персональних каналів обміну текстовими повідомленнями між двома користувачами.

Для відправлення текстового повідомлення в загальний канал обміну текстовими повідомленнями # *Main* необхідно у полі «**2**» ввести призначене для відправлення текстове повідомлення і натиснути клавішу ENTER. Текст повідомлення буде відправлено усім зареєстрованим користувачам і відображено в полі «**1**».

Для відправлення текстового повідомлення конкретному користувачеві необхідно в полі «**З**» вибрати ім'я цього користувача зі списку, виділити його правою кнопкою миші та у меню, яке з'явиться, вибрати пункт «**Сообщение**». Після чого у вікні «**Текст сообщения:**» (рис. Л.7.2) набрати текст повідомлення і натиснути «**Отправить**».

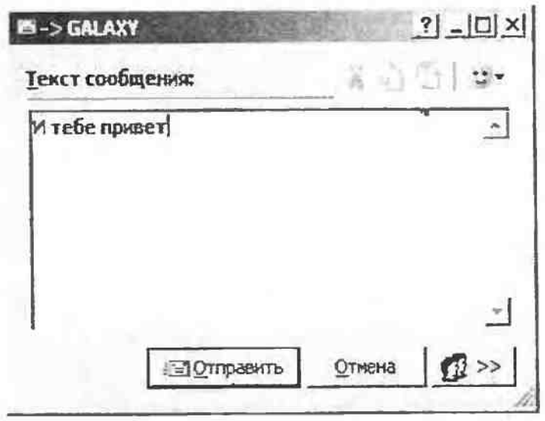


Рис. Л.7.2. Вікно «**Текст сообщения**»

У полі загального каналу обміну текстовими повідомленнями # *Main* буде відображено звіт про час одержання цього повідомлення. У користувача, що одержав це повідомлення, з'явиться вікно «**Полученное сообщение:**» (рис. Л.7.3), яке містить текст повідомлення.

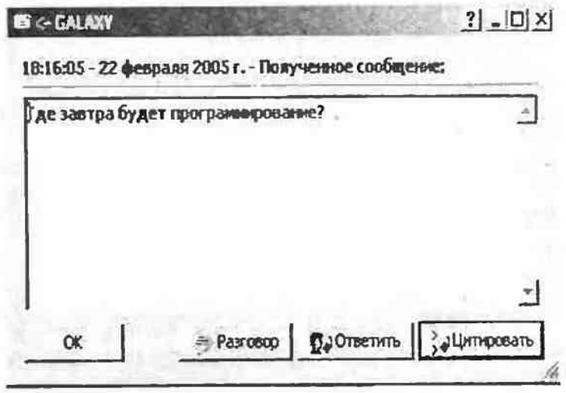


Рис. Л.7.3. Вікно «**Полученное сообщение:**»

Доступні пункти:
— «Разговор»;
— «Ответить»;
— «Цитировать».

Пункт «**Разговор**» дає змогу створювати персональний канал обміну текстовими повідомленнями між двома користувачами. При цьому в полі загального каналу обміну текстовими повідомленнями # *Main* буде відображатися службова інформація про прийом текстових повідомлень і час їх отримання.

При виборі пункту «**Ответить**» повідомлення буде відправлене тому користувачеві, від якого було отримано повідомлення.

Пункт «**Цитировать**» аналогічний пунктові «**Ответить**», але в цьому випадку повідомлення міститиме в одному вікні і цитоване попереднє повідомлення.

Текст повідомлення вводиться в полі «**2**».

Уміст каналу обміну текстовими повідомленнями може бути очищено. Для цього потрібно натиснути праву кнопку миші та в меню, що з'явиться, вибрати пункт «**Очистить чат**».

Для закриття каналу обміну текстовими повідомленнями треба натиснути праву кнопку миші й в меню, яке з'явиться, вибрати пункт «**Оставить**».

У панелі інструментів відображається режим, у якому знаходиться користувач (рис. Л.7.4).

Рис. Л.7.4. Режим у якому знаходиться користувач

Ця опція також доступна за командою «**Разговор**» ⇒ «**Режим**...». Можливі режими:
– «Активен»;
– «Не беспокоить»;
– «Ушел»;
– «Отключенный».

Якщо користувач активний, він здатен приймати і відправляти повідомлення. Такий режим є звичайним для обміну текстовими повідомленнями. Для зміни режиму роботи користувача потрібно вибрати інший режим, наприклад, «**Не беспокоить**». У вікні «**Автоответ в режиме не беспокоить**» (рис. Л.7.5), треба ввести автовідповідь — текст, що буде відображатись автоматично в каналі обміну текстовими повідомленнями при одержанні текстових повідомлень. При цьому в поле «**З**», у якому відображається список зареєстрованих користувачів, зміниться режим даного користувача. Аналогічно — з режимами «**Ушел**» і «**Отключенный**».

Пункт «**Каналы**» у панелі інструментів (рис. Л.7.6) дозволяє здійснювати керування каналами обміну текстовими повідомлення-

ми, наприклад, додавати і видаляти їх. Ця опція також доступна за командою **«Разговор» ⇒ «Каналы…»** або при натисканні клавіші F2.

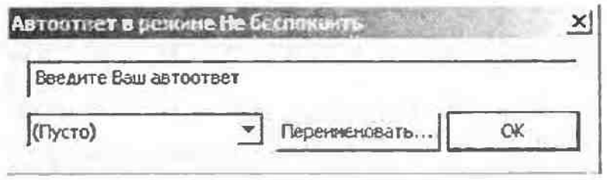


Рис. Л.7.5. Режим **«Не беспокоить»**.

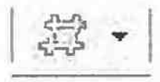


Рис. Л.7.6. Пункт **«Каналы»** у панелі інструментів

Тему обговорення в каналі обміну повідомленнями можна змінити, для цього потрібно виконати команду **«Разговор» ⇒ «Изменение темы»**. З'явиться вікно, в якому вказується нова тема обговорення, а в каналі обміну повідомленнями з'явиться повідомлення про це.

Програма *Vypress Chat* дає змогу автоматично вести протокол обміну текстовими повідомленнями в кожному каналі обміну повідомленнями, записувати і переглядати усі відправлені й отримані повідомлення.

Архів обміну текстовими повідомленнями (рис. Л.7.7) можна переглянути, натиснувши лівою кнопкою миші на цьому пункті, командою **«Разговор» ⇒ «Архив сообщений»** або натиснувши клавішу F11.

Рис. Л.7.7. Архів обміну текстовими повідомленнями

Завдання

1. Запустити на віртуальних комп'ютерах програму *Vypress Chat*. Занести до звіту вміст поля 3, де відображаються зареєстровані користувачі. Після цього кожен користувач змінює свій псевдонім. Новий псевдонім і службову інформацію з каналу *# Main* занести до звіту.

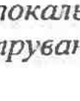
2. Кожний з користувачів відправляє текстове повідомлення в загальний канал обміну повідомленнями *# Main*. Занести до звіту текст повідомлень і службову інформацію з каналу *# Main*.

3. За домовленістю користувачі відправляють по одному текстовому повідомленню обраному заздалегідь користувачеві, потім наступним. Користувач, що одержав повідомлення, відправляє повідомлення у відповідь, використовуючи пункти **«Ответить»** і **«Цитировать»**. Занести до звіту зміст каналу *# Main*.

4. За домовленістю, вибравши пункт **«Разговор»**, створити персональний канал обміну текстовими повідомленнями між двома користувачами, відправити один одному два текстові повідомлення. Занести до звіту текстові повідомлення і службову інформація з каналу *# Main*, після чого закрити всі канали, крім *# Main*.

5. При обміні текстовими повідомленнями в загальному каналі *# Main* кожен користувач змінює свій режим роботи (активний, не турбувати, пішов і відключений). Занести до звіту службову інформацію з каналу *# Main*.

6. Переглянути файл протоколу обміну текстовими повідомленнями. Занести до звіту, яка інформація там відображається.



Питання для самоперевірки

1. Призначення персонального каналу обміну текстовими повідомленнями в програмі *Vypress Chat*.

2. Функціональне призначення каналу обміну текстовими повідомленнями програми *Vypress Chat*.

3. Які опції доступні користувачу, зареєстрованому в програмі *Vypress Chat*, при одержанні персонального текстового повідомлення?

4. У яких режимах може працювати користувач, зареєстрований в програмі *Vypress Chat*?

5. Для чого призначена програма *Vypress Chat*?

6. Основні дії при створенні персонального каналу обміну текстовими повідомленнями програми *Vypress Chat*.

7. Службова інформація, яка відображається у вікні каналу обміну текстовими повідомленнями програми *Vypress Chat*.

8. Особливості роботи зареєстрованого користувача програми *Vypress Chat*.

ЛАБОРАТОРНА РОБОТА № 8

Дослідження програми управління мережними ресурсами *NET BLOCK PRO RE*

Мета роботи:

вивчення спеціалізованого програмного забезпечення для роботи в локальних комп'ютерних мережах, призначеного для адміністрування мережних ресурсів комп'ютера.

Короткі теоретичні відомості

Програма *Net Block Pro* RE призначена для проведення операцій над списком мережних ресурсів комп'ютера і їх оперативного перерозподілу.

Важливою особливістю цієї програми є можливість оперативного оповіщення про факт підключення користувача з іншого комп'ютера до мережного ресурсу локальної мережі або мережі *Internet*.

Програма має можливості:

— виведення інформації про доступні мережні ресурси, підключених до цього комп'ютера користувачів і відкриті файли;

— додавання, видалення, редагування мережних ресурсів, у тому числі схованих;

— відключення користувачів від мережного ресурсу;

— закриття відкритого користувачем файла;

— керування параметрами безпеки для доступних мережних ресурсів;

— блокування доступу до ресурсу;

— контролю підключення користувачів з інших комп'ютерів до мережних ресурсів комп'ютера і виведення оперативної інформації про мережні підключення;

— ведення лог-файла підключень користувачів (історія підключень);

— вибору фільтрів для виведеної інформації.

Програма *Net Block Pro* RE складається з двох частин: **«Агент»** і **«Менеджер»**.

«Агент» запускається в процесі завантаження комп'ютера і здійснює контроль над підключеннями користувачів до доступних мережних ресурсів. При підключенні користувача до якого-небудь мережного ресурсу **«Агент»** може вивести вікно з повідомленнями (рис. Л.8.1), подати звуковий сигнал і записати повідомлення у лог-файл.

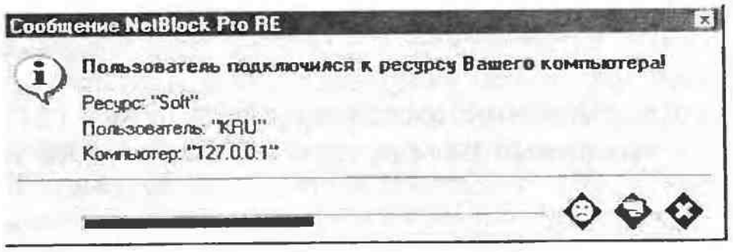


Рис. Л.8.1. Вікно повідомлень при підключенні користувача до мережного ресурсу

Доступні опції:

— відключення користувача від мережних ресурсів комп'ютера;

— запуск програмної частини «Менеджер»;

— вихід з програмної частини **«Агент»**.

«Менеджер» дозволяє керувати мережними ресурсами комп'ютера і має три режими роботи:

«Ресурсы» — висновок інформації про мережні ресурси;

«Пользователи» — висновок інформації про підключених користувачів;

«Файлы» — висновок інформації про відкритий користувачами файл.

Розглянемо ці режими роботи докладніше.

1. Режим **«Ресурсы»** (рис. Л.8.2).

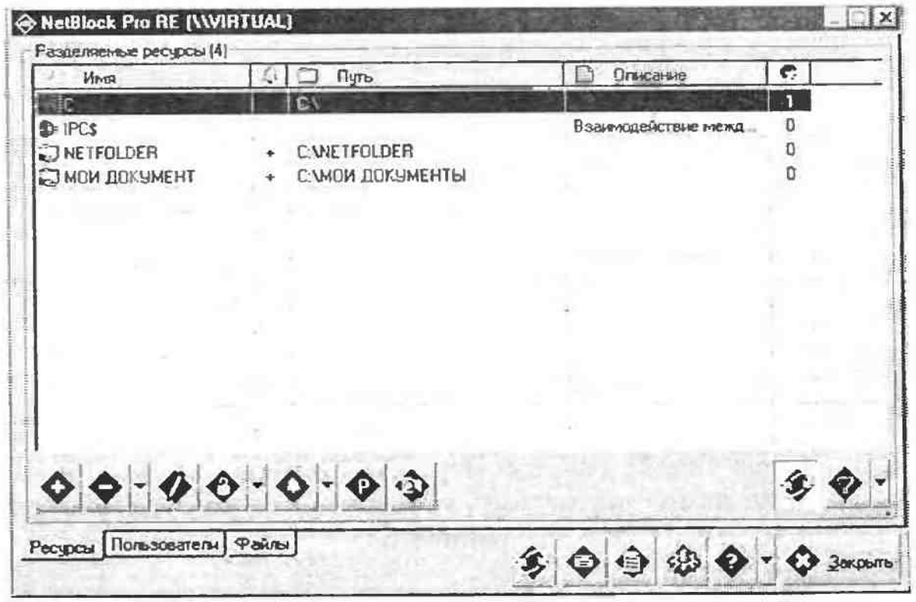


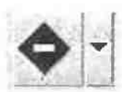
Рис. Л.8.2. Вікно **«Ресурсы»** програмної частини **«Менеджер»**

У полі списку зазначено мережні ресурси комп'ютера, їхні імена, шлях до ресурсу, його опис.

У нижній частині вікна містяться піктограми — елементи керування.

— додавання мережного ресурсу.

Після натискання на цю піктограму з'являється вікно **«Добавление нового ресурса»**. Для додавання мережного ресурсу потрібно натиснути правою кнопкою миші на позначку **«Папка»** та в каталогах вибрати об'єкт, до якого треба одержати доступ, і натиснути **«ОК»**. Відтак у вікні **«Добавление нового ресурса»** у відповідних полях буде зазначено шлях до мережного ресурсу, його ім'я і може бути доданий його опис (необов'язковий параметр). Можна зробити цей ресурс схованим. Після вказівки необхідних параметрів потрібно натиснути **«ОК»**.



— видалення мережного ресурсу. При звертанні до цієї піктограми з'являється вікно **«Подтверждение удаления»**. Для підтвердження видалення слід натиснути **«Да»**.

Крім того, доступні додаткові можливості:
— видалення всіх блокованих ресурсів;
— видалення всіх не схованих ресурсів.

— редагування мережного ресурсу. Можна змінити шлях до об'єкта, змінити його ім'я, зробити цей мережний ресурс схованим.

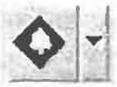
Підпункт **«Еще»** дозволяє редагувати властивості об'єкта (атрибути і тип доступу, можливість установлення пароля для доступу).

— блокування / розблокування ресурсу.

Якщо в полі списку вибрати мережний ресурс і натиснути лівою кнопкою миші на цю піктограму, мережний ресурс буде заблоковано і доступ до нього стане неможливий.

Додаткові можливості:
— блокувати всі мережні ресурси;
— розблокувати всі мережні ресурси.

— зміна системних настроювань мережного ресурсу (зміна атрибутів об'єкта і типу доступу до нього).



— установлення / відхилення контролю за підключеннями до мережного ресурсу (якщо контроль включено, при звертанні до ресурсу буде виводитися відповідне повідомлення (див. рис.Л.8.2)).

— показати шляхи до мережного ресурсу. При натисканні правої кнопки миші на цій піктограмі з'являється вікно **«Проводник»**, де відображаються доступні мережні ресурси.

Додаткові можливості
— контролювати всі ресурси (контроль включено);
— не контролювати всі ресурси (контроль включено).

— включення / виключення автооновлення.

— включення / відключення фільтрації інформації (вказує, які мережні ресурси у вікні програмної частини **«Менеджер»**).

Унизу цього вікна розташовані такі піктограми:

— примусове відновлення інформації;

— показати лог-файл;

— настроювання програми *Net Block Pro* RE.

***2. Режим* «Пользователь»** (рис. Л.8.3)

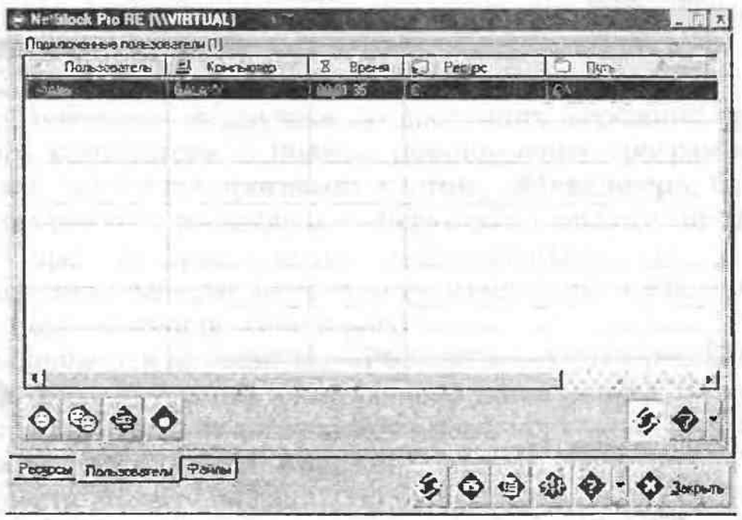


Рис. Л.8.3. Вікно **«Пользователи»** програмної частини **«Менеджер»**

У полі списку відображається інформація про ім'я підключеного користувача, час підключення, ім'я мережного ресурсу, до якого здійснено підключення, і шлях до цього мережного ресурсу.

Доступні опції:

— відключення користувача від мережного ресурсу;

— відключення всіх користувачів від мережного ресурсу;

— підключення до віддаленого комп'ютера. При натисканні правої кнопки миші на цій піктограмі з'являється вікно **«Проводник»**, за допомогою якого можна підключитися до віддаленого комп'ютера в локальній мережі;

— блокування / розблокування ресурсу. Натискання лівої кнопки миші дозволяє оперативно блокувати / розблокувати доступ до обраного мережного ресурсу.

3. Режим **«Файлы»** (рис. Л.8.4)

У полі списку відображається інформація про те, які файли відкриті віддаленими користувачами, ім'я користувача, що відкрив файли, ім'я комп'ютера, з якого здійснюється доступ до цих файлів, а також ім'я доступного мережного ресурсу і повний шлях до нього.

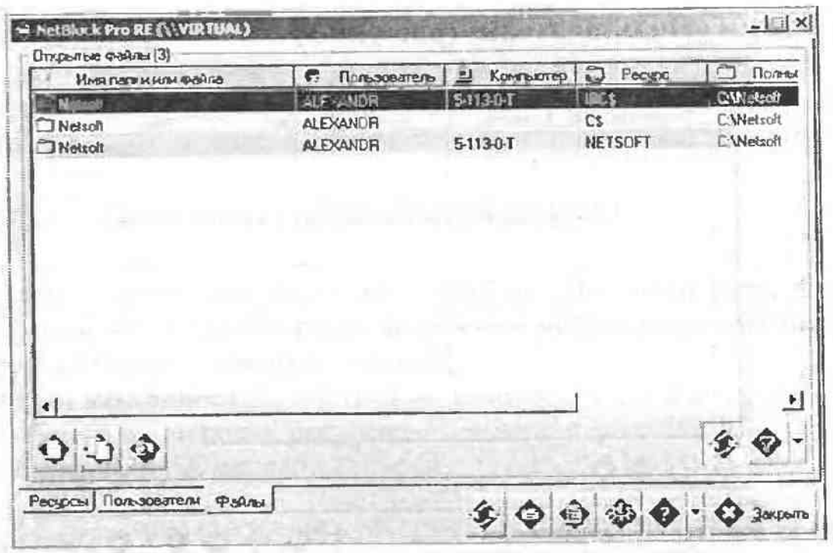


Рис. Л.8.4. Вікно **«Файлы»** програмної частини **«Менеджер»**

Доступні опції:

— закрити файл, відкритий віддаленим користувачем;

— закрити усі файли, відкриті віддаленим користувачем;

— показати шлях доступного мережного ресурсу. При натисканні лівої кнопки миші викликається «Провідник», що вказує шлях до мережного ресурсу.

Завдання

1. На кожному віртуальному комп'ютері створити 3—4 мережні папки, у які потрібно помістити файли (**.txt*, **.html* та інші). Представити повний мережний доступ до цих папок. Надати доступ до всього диска С: віртуального комп'ютера.

Перед виконанням цієї операції треба визначити мережне ім'я віртуального комп'ютера (**«Пуск»** ⇒ **«Настройка»** ⇒ **«Панель управления»** ⇒ **«Сеть»** ⇒ **«Идентификация»**). Усі віртуальні комп'ютери мають позначку «*Virtual*».

За допомогою піктограми **«Сетевое окружение»** реального комп'ютера звернутися до мережного ресурсу віртуального комп'ютера.

2. При звертанні до мережного ресурсу віртуального комп'ютера переключитися на віртуальний комп'ютер. За допомогою меню програми *Connectix Virtual* PC, виконавши команду ***File ⇒ Pause,*** призупинити роботу емулятора, коли з'явиться повідомлення про підключення віддаленого користувача. Занести до звіту повідомлення програмної частини **«Агент»**, після чого відновити роботу програми, виконавши команду ***File ⇒ Resume***.

3. За допомогою піктограми **«Сетевое окружение»** реального комп'ютера повторно звернутися до доступних мережних ресурсів віртуального комп'ютера. З появою повідомлення програмної частини **«Агент»** запустити програмну частину **«Менеджер»**. Після запуску переключитися на закладку **«Ресурсы»** і занести до звіту інформацію про ресурси, потім переключитися на закладку **«Пользователи»** і занести до звіту інформацію про віддаленого користувача, після чого відключити його.

4. Переключитися на закладку **«Ресурсы»** і додати новий мережний ресурс (будь-яку папку віртуального комп'ютера). За допомогою піктограми **«Сетевое окружение»** реального комп'ютера перевірити, чи з'явився новий мережний ресурс. При появі нового ресурсу занести до звіту операції, які були проведені для цього.

5. Будучи на закладці **«Ресурсы»** видалити мережний ресурс, створений у пункті 4. Перевірити за допомогою піктограми **«Сетевое окружение»** реального комп'ютера, чи вилучений цей мережний ресурс. Занести до звіту, які операції були для цього виконані.

6. Будучи на закладці **«Ресурсы»** заблокувати мережний ресурс. Перевірити результат за допомогою піктограми **«Сетевое окружение»** реального комп'ютера. Результати занести до звіту.

7. За допомогою піктограми **«Сетевое окружение»** на реальному комп'ютері запустити на віртуальному комп'ютері програму *Windows Media Player*, що міститься на диску (C:\Program *files*\ *Windows Media Player*). Після запуску цієї програми переключитися на віртуальний комп'ютер, перейти на закладку **«Файлы»** програми *Net Block* PRO RE і занести до звіту інформацію з поля списку, після чого закрити цей файл.

8. Переглянути зміст лог-файла і записати його до звіту.

Питання для самоперевірки

1. У чому різниця між програмними частинами **«Менеджер»** і **«Агент»** у програмі *Net Block*?

2. Яка службова інформація записується в лог-файл програми *Net Block*?

3. Охарактеризувати режими роботи програмної частини **«Менеджер»** програми *Net Block*.

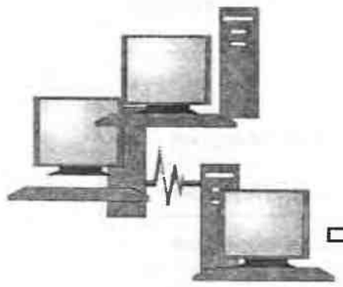
4. Режими роботи програмної частини **«Агент»** програми *Net Block*.

5. Які дії доступні над мережними ресурсами в режимі **«Ресурсы»** програмної частини **«Менеджер»** програми *Net Block*?

6. Яка службова інформація відображається програмною частиною **«Менеджер»** програми *Net Block* в режимі **«Ресурсы»**?

7. Яка службова інформація відображається програмною частиною **«Менеджер»** програми *Net Block* в режимі **«Файлы»**?

8. Яка службова інформація відображається програмною частиною **«Менеджер»** програми *Net Block* в режимі **«Пользователи»**?



Домашнє завдання

Домашнє завдання виконується для закріплення та поглиблення теоретичних знань та вмінь, набутих студентом під час теоретичних і лабораторних занять, і складається з двох частин: практичної та реферативної.

Розробка, оформлення та захист домашнього завдання здійснюється студентом в індивідуальному порядку відповідно до варіанта.

Практична частина

Тема: «Проектування контролера зовнішнього пристрою з вбудованим контролем».

Завдання на практичну частину вибирається згідно з номером варіанта (табл. 1).

Таблиця 1

№ варіанта	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Код завдання	2	1	1	4	5	3	2	3	1	5	4	2	3	4	5	1
	A	C	G	C	D	F	D	C	E	F	D	G	E	F	C	B

17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
3	2	1	1	4	5	3	2	3	1	5	4	2	3	4	5	1
D	A	F	E	A	F	B	C	G	E	B	E	A	D	G	A	B

Код завдання містить:

1. Код контролера та варіант завдання на реферативну частину задається цифрою (табл. 2).

Таблиця 2

Код	Тип пристрою
1	Контролер паралельної синхронної передачі
2	Контролер паралельної асинхронної передачі
3	Контролер послідовної синхронної передачі
4	Контролер послідовної асинхронної передачі
5	Контролер прямого доступу до пам'яті

2. Код методу контролю задається буквою (табл. 3).

Таблиця 3

Код	Метод контролю
A	Дублювання
B	Троювання
C	Формування контрольного розряду до парності
D	Формування контрольного розряду до непарності
E	Простий код Хеммінга
F	Модифікований код Хеммінга
G	Групового кодування

У схемі використовується 16-розрядна шина даних.

Реферативна частина

Згідно з номером варіанта з таблиць 4—6 вибирається тип топології комп'ютерних мереж, мережне обладнання та середовище передачі даних, яке треба детально описати.

Завдання на реферативну частину

1. Топологія комп'ютерних мереж. Тип мережі, його опис, загальні характеристики. Принцип функціонування мереж з цією топологією. Переваги та недоліки (табл. 4).

Таблиця 4

Тип	Варіант
Шинна топологія	1
Зіркоподібна топологія	2
Кільцева топологія	3
Змішана (шинно-зіркоподібна топологія)	4
Змішана (зіркоподібно-кільцева топологія)	5

2. Мережне обладнання, що використовується в комп'ютерних мережах.

Загальна характеристика та принцип дії (табл. 5).

Таблиця 5

Тип	Варіант
Концентратор	1
Повторювач (*repeater*)	2
Комутатор (*switch*)	3
Міст (*dredge*)	4
Хаб (*hub*)	5

3. Середовище передачі даних. Загальна характеристика, пропускна здатність, переваги та недоліки (табл. 6).

Таблиця 6

Тип	Варіант
Безпровідна	1
Кабелі на основі витої пари	2
Товсті коаксіальні кабелі	3
Тонкі коаксіальні кабелі	4
Оптоволоконні кабелі	5

Зміст звіту

1. Титульна сторінка.
2. Практична частина:
а) вступ;
б) технічне завдання;
в) опис роботи контролера;
г) опис методу контролю;
д) проектування контролера зі схемою вбудованого контролю:
— проектування структурної схеми (робота контролера на вхід та вихід інформації);
— проектування принципової схеми (робота контролера на вхід та вихід інформації).
3. Реферативна частина.
4. Висновок.
5. Список літератури.

Приклад титульної сторінки звіту

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Інститут комп'ютерних технологій

Факультет комп'ютерних систем

ЗВІТ
домашнього завдання з дисципліни
«Експлуатація комп'ютерних систем та мереж»

Виконав студент 514 гр. Шевченко О.П.

Перевірив Василенко О.О.

Київ-2007

ДОДАТКИ

ДОДАТОК 1

ПРОГРАМА ЕМУЛЯЦІЇ ПЕРСОНАЛЬНОГО КОМП'ЮТЕРА CONNECTIX PC 5.1

Програма емуляції дозволяє створювати віртуальний комп'ютер, установлювати на нього операційні системи: DOS, *Windows* 3x/9x/ME/2000/XP, *Linux*, *FreeBSD*, *OS*/2, *Solaris* і деякі інші, а також програмне забезпечення для цих операційних систем. Крім того, є можливість створювати віртуальну комп'ютерну мережу між реальним і віртуальним комп'ютером і працювати з нею, як з реальною.

При запуску програми з'являється вікно (рис. Д. 1.1), в якому знаходиться список доступних віртуальних комп'ютерів.

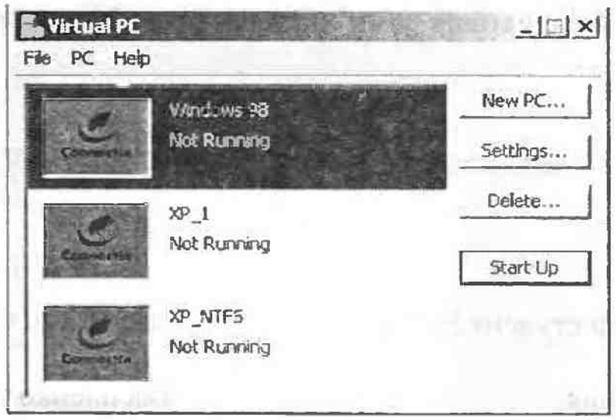


Рис. Д. 1.1. Вікно доступних віртуальних комп'ютерів *Virtual* PC

Пункти меню:

— *New* PC — створення нового віртуального комп'ютера;
— *Settings* — властивості вибраного віртуального комп'ютера;
— *Delete* — видалення вибраного віртуального комп'ютера;
— *Start Up* — увімкнення обраного віртуального комп'ютера.

Пункти *File* та PC майже повністю дублюють це меню за винятком таких дій:

— команда *File* ⇒ *Virtual Disk Wizard* надає вихід до майстра створення нових жорстких дисків;

— команда *PC* ⇒ *Properties* надає вихід до службової інформації про обрану віртуальну машину.

Після ввімкнення обраного віртуального комп'ютера в окремому вікні програми завантажиться операційна система, встановлена на цьому віртуальному комп'ютері (рис. Д. 1.2).

Меню програми:
PC — керування віртуальним комп'ютером;
Edit — настроювання параметрів віртуального комп'ютера;
CD — керування приводом CD-ROM;
Floppy — керування дисководом.

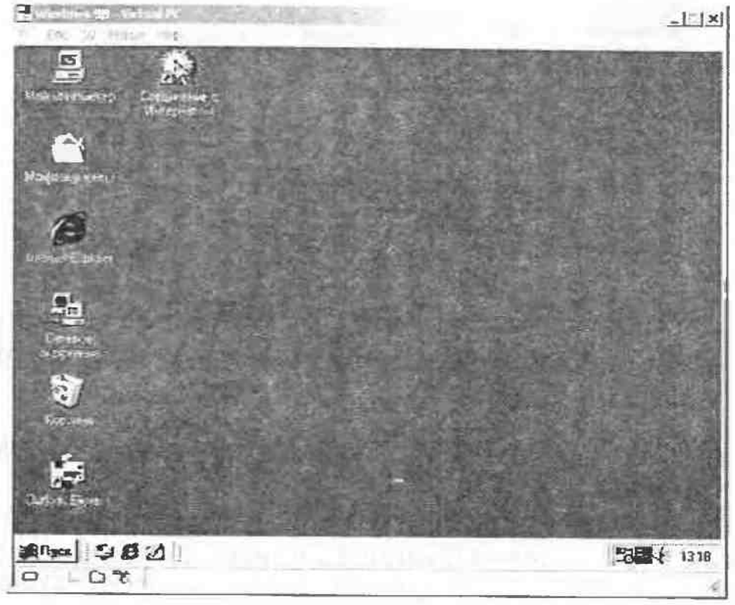


Рис. Д.1.2. Вікно інтерфейсу певної операційної системи

— індикатори та кнопки настроювання параметрів відповідно жорстких дисків, приводу CD-ROM, дискет, доступних мережних папок і мережного адаптера.

Розглянемо докладніше пункти меню:

1) PC:

— *Enable Full Screen* — перехід віртуального комп'ютера у повноекранний режим роботи;

— *Type Ctrl+Alt+Del* — натискання на віртуальному комп'ютері комбінації клавіш *Ctrl+Alt+Del*;

— *Pause / Resume* — призупинення / поновлення роботи віртуального комп'ютера;

— *Reset* — перезавантаження віртуального комп'ютера;

— *Shut Down* — вимикання віртуального комп'ютера;

— *Install / Update Additions* — інсталяція / оновлення драйверів емулятора;

— *Properties* — службова інформація про вибрану віртуальну машину.

2) *Edit:*

Settings — настроювання віртуального комп'ютера (рис. Д. 1.3).

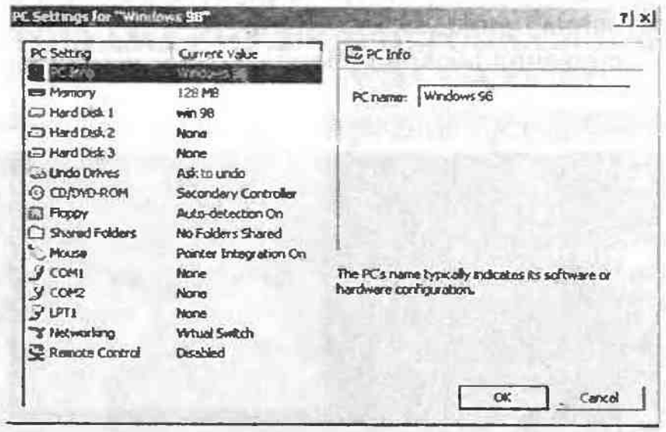


Рис. Д. 1.3. Вікно настроювання віртуального комп'ютера

3) CD:

— *Capture Host Drive* — підключення привода CD-ROM реального комп'ютера;

— *Capture Image* — підключення образу віртуального CD-диска;

— *Release Disk* — демонтування CD-диска або його образу;

— *Eject Host Drive* — відкриття привода CD-ROM реального комп'ютера.

4) *Floppy:*

— *Capture Host Drive A:* — підключення дисковода реального комп'ютера;

— *Capture Image* — підключення образу віртуального дисковода;

— *Release Disk* — демонтування дискети або її образу.

Для роботи віртуального комп'ютера у віртуальній комп'ютерній мережі необхідно зробити такі настроювання емулятора:

— в пункті *Setting* вибрати пункт *Virtual Switch*;

— установити режим *Local, host and external;*

— натиснути «**ОК**».

При вимкненні віртуального комп'ютера з'явиться меню, яке дозволяє зберегти або відмовитися від усіх змін, які були зроблені під час останнього сеансу роботи (рис. Д. 1.4).

При виборі пункту *Commit all hard drive changes* зміни, які були зроблені в останньому сеансі роботи на віртуальному комп'ютері, набудуть чинності. При виборі пункту *Undo all hard drive changes* – всі зміни, які були зроблені в останньому сеансі, будуть відмінені. Після цього віртуальний комп'ютер буде вимкнено.

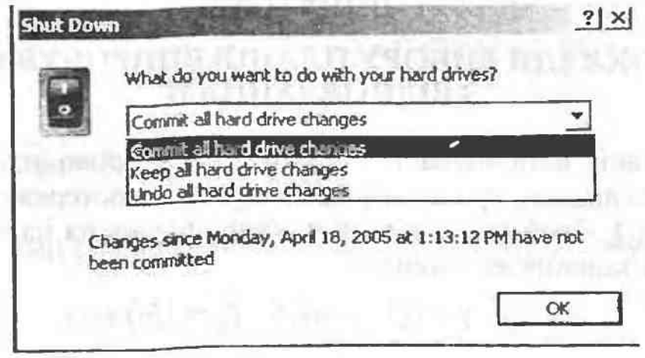


Рис. Д. 1.4. Вікно меню зберігання змін

ДОДАТОК 2

ПРИКЛАДИ ВИБОРУ ПЛАНІВ ВИПРОБУВАНЬ З ВІДНОВЛЕННЯМ

Вибір планів випробувань з відновленням проводиться методом послідовного аналізу при безперервному часі спостереження.

Приклад 1. Зробити розрахунок випробувань на надійність двох об'єктів при заданих величинах:

$$\alpha = 0{,}1;\ \beta = 0{,}1;\ r = 1{,}5;\ T_0 = 150 \text{ год.}$$

а) Середню тривалість випробувань M_T при $T = T_0$ визначаємо за формулою

$$M_{T_0}\{t\} = \frac{h_0 T_0}{n}$$

і розраховуємо її з використанням значення середньої тривалості випробувань h_0, отриманих у табл. Д.2.1:

$$M_{T_0}\{t\} = \frac{18{,}6 \cdot 150}{2} = 1395 \text{ год.}$$

Середню тривалість випробувань при $T = T_1$ визначаємо за аналогічною формулою

$$M_{T_1}\{t\} = \frac{h_1 T_1}{n},$$

де h_1 визначається з табл. Д.2.2, і також розраховуємо:

$$M_{T_1}\{t\} = \frac{24{,}43 \cdot 100}{2} = 1221{,}5 \text{ год.}$$

б) За допомогою формул границь області відповідності

$$\underline{k}(t) = \frac{n(r-1)}{T_0 \ln r} t + \frac{\ln\left[\beta(1-\alpha)^{-1}\right]}{\ln r}$$

та границі області невідповідності

$$\overline{k}(t) = \frac{n(r-1)}{T_0 \ln r} t + \frac{\ln \tilde{A}}{\ln r}$$

знаходимо відповідні рівняння $\underline{k}(t)$ та $\overline{k}(t)$ з урахуванням значень $\ln \tilde{A}$ — з табл. Д.2.3, а $\ln\left[\beta(1-\alpha)^{-1}\right]$ — з табл. Д.2.4:

$$\underline{k}(t) = \frac{2(1{,}5-1)}{150 \ln 1{,}5} t + \frac{-2{,}1972}{\ln 1{,}5} = 0{,}0164t - 5{,}41;$$

$$\overline{k}(t) = 0{,}0164t + \frac{2{,}062}{0{,}4055} = 0{,}0164t + 5{,}08.$$

За отриманими рівняннями будуємо графік (рис. Д.2.1).

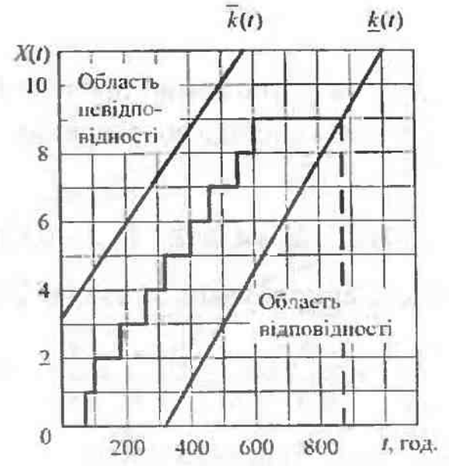


Рис. Д.2.1

Випробування продовжуємо доти, поки виконується умова

$$0{,}0164t + 5{,}08 > x(t) > 0{,}0164t - 5{,}41.$$

Випробування варто припинити в момент часу $t = \tilde{t}$, коли виконується одна з умов:

$$x(\tilde{t}) \le 0{,}0164\tilde{t} - 5{,}41 \text{ або } x(\tilde{t}) \ge 0{,}0164\tilde{t} + 5{,}08.$$

При цьому в першому випадку об'єкт вважається відповідним заданим вимогам, у другому — ні.

Найменша тривалість випробувань з ухваленням позитивного рішення може дорівнювати 330 год за умови, що за 330 год не виникне жодної відмови.

Якщо, наприклад, 9-та відмова виникла на 600-й годині випробувань, то випробування після відновлення виробу продовжуються і можуть бути припинені на 879-й годині з ухваленням позитивного рішення за умови, що кількість відмов залишилася рівною 9 (див. рис. Д.2.1).

Приклад 2. Зробити розрахунок випробувань на надійність трьох виробів за таких заданих параметрів:

$$\alpha = 0{,}1;\ \beta = 0{,}15;\ q_0 = 0{,}1;\ \frac{q_1}{q_0} = 2{,}5.$$

Насамперед з табл. Д.2.5 за заданими $q_0 = 0{,}1$ і $\frac{q_1}{q_0} = 2{,}5$ знаходимо $r = 2{,}73$ і $\frac{t_{бр}}{T_0} = 0{,}105$.

а) З табл. Д.2.1—Д.2.3 за допомогою правила пропорційних частин (лінійної інтерполяції) знаходимо значення h_0, h_1, $\ln \tilde{A}$ для $r = 2{,}73$ при $\alpha = 0{,}1$ і $\beta = 0{,}15$:

$$h_0 = 2{,}011;\quad h_1 \approx 4{,}363;\quad \ln \tilde{A} = 1{,}810.$$

б) Середня тривалість випробувань за умови $q = q_0$

$$M_{T_0}\{t\} = \frac{h_0 T_0}{n} = \frac{h_0 t_{бр}}{n \cdot 0{,}105} = \frac{2{,}011 \cdot 150}{3 \cdot 0{,}105} \approx 958 \text{ год.}$$

Середня тривалість випробувань за умови $q = q_1$

$$M_{T_1}\{t\} = \frac{h_1 T_0}{rn} = \frac{h_1 t_{бр}}{rn \cdot 0{,}105} = \frac{4{,}363 \cdot 150}{2{,}73 \cdot 3 \cdot 0{,}105} \approx 760 \text{ год.}$$

в) Рівняння границі областей відповідності

$$\underline{k}(t) = \frac{n(r-1)}{T_0 \ln r} + \frac{\ln\left[\beta(1-\alpha)^{-1}\right]}{\ln r} =$$

$$\frac{3(2{,}73-1)\cdot 0{,}105}{150 \ln 2{,}73} t + \frac{-1{,}792}{\ln 2{,}73} = 0{,}0036t - 1{,}784.$$

Рівняння границі області невідповідності

$$\bar{k}(t) = 0{,}0036t + \frac{\ln \tilde{A}}{\ln r} = 0{,}0036t + \frac{1{,}8100}{1{,}0043} = 0{,}0036t + 1{,}802.$$

За отриманими рівняннями будуємо графік (рис. Д.2.2).
Випробування продовжуємо доти, поки виконується умова

$$0{,}0036t + 1{,}802 > x(t) > 0{,}0036t - 1{,}784.$$

Випробування варто припинити в той момент часу $t = \tilde{t}$, коли виконується одна з умов:

$$x(\tilde{t}) \le 0{,}0036\tilde{t} - 1{,}784 \text{ або } x(\tilde{t}) \le 0{,}0036\tilde{t} - 1{,}784.$$

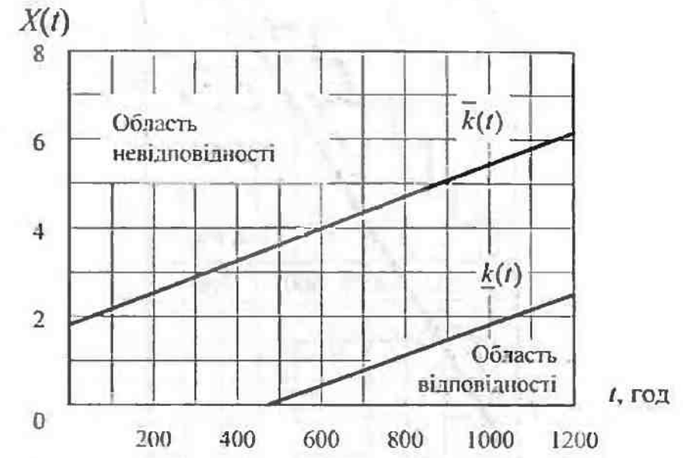


Рис. Д.2.2

При цьому в першому випадку об'єкти вважаються відповідними заданим вимогам щодо надійності, у другому — ні.

Найменша тривалість випробувань з ухваленням позитивного рішення може дорівнювати 490 год за умови, що за 490 год не виникло жодної відмови.

Приклад 3. Проводяться випробування на надійність об'єкта при заданих значеннях $\lambda = 6{,}67 \cdot 10^{-3} \frac{1}{\text{год}}$; $\alpha = 0{,}1$; $\beta = 0{,}1$; $r = 1{,}5$ і розрахункових величинах $I = 5{,}08$; $K = -5{,}41$.

Реалізація процесу відмовлень $x(t)$ тривалий час знаходиться між границями (рис. Д.2.3). З огляду на те, що функція $x(t)$ має тенденцію наближення до нижньої границі, приймається рішення про те, що об'єкт відповідає заданим вимогам надійності. Така операція приведе до нових значень α і β, які необхідно оцінити.

Насамперед визначаємо нове значення K. Для цього через крапку $(\tilde{t}, x(\tilde{t}))$ — кінець спостереженої траєкторії — проводимо лінію, рівнобіжну розрахунковій, до перетинання з віссю ординат і знаходимо $K = -4{,}4$.

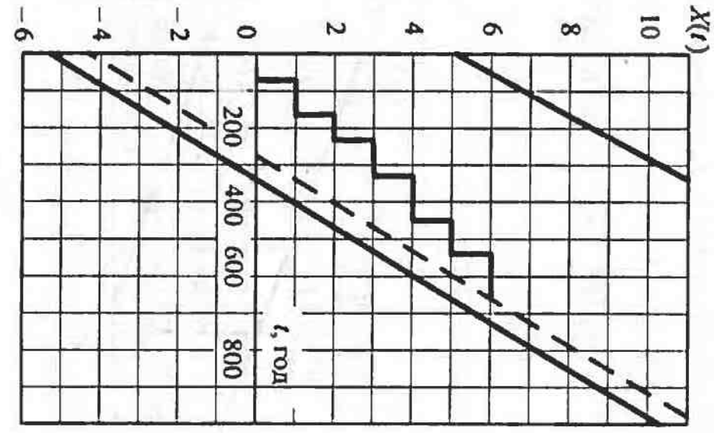


Рис. Д.2.3

Потім, підставляючи значення $I = 5{,}08$ і $K = -4{,}4$ у вирази визначення нових значень ризику постачальника α та замовника β

$$\frac{1-r^{K}}{r^{I+1}-r^{K}} \le \alpha \le \frac{1-r^{K}}{r^{I}-r^{K}}; \quad \frac{r^{i}-1}{r^{I-K}-1} \le \beta \le \frac{r^{I+1}-1}{r^{I-K+1}-1}$$

дістаємо такі інтервали:

$$0{,}0719 \le \alpha \le 0{,}1085; \quad 0{,}1498 \le \beta \le 0{,}1559.$$

ЗНАЧЕННЯ СЕРЕДНЬОЇ ТРИВАЛОСТІ ВИПРОБУВАНЬ h_0

Таблиця Д. 2.1

Ризик постачальника α	Ризик замовника β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
$r = 1{,}2$										
0,05	149,90	112,80	91,18	75,89	64,08	54,48	46,41	39,46	33,37	27,98
0,10	134,40	99,44	79,12	64,81	53,82	44,93	37,50	31,15	25,65	20,83
0,15	120,60	87,70	68,69	55,37	45,19	37,01	30,23	24,49	19,56	15,31
0,20	107,90	77,09	59,39	47,06	37,69	30,22	24,07	18,95	14,60	10,91
0,25	96,02	67,38	50,99	39,64	31,09	24,33	18,83	14,30	10,53	7,41
0,30	84,95	58,42	43,34	32,98	25,24	19,19	14,34	10,41	7,22	4,67
0,35	74,56	50,14	36,37	26,99	20,06	14,72	10,52	7,20	4,59	2,60
0,40	64,77	42,48	30,02	21,62	15,51	10,86	7,33	4,61	2,58	1,16
0,45	55,60	35,42	24,26	16,85	11,55	7,64	4,73	2,60	1,16	—
0,50	46,99	28,92	19,07	12,65	8,16	4,96	2,69	1,18	—	—
$r = 1{,}3$										
0,05	70,42	52,99	42,83	35,65	30,10	25,59	21,80	18,54	15,68	13,15
0,10	63,14	46,71	37,17	30,45	25,28	21,11	17,62	14,64	12,05	9,78
0,15	56,64	41,20	32,27	26,01	21,23	17,39	14,20	11,51	9,12	7,19
0,20	50,67	36,22	27,90	22,11	17,71	14,21	11,32	8,90	6,86	5,13
0,25	45,11	31,66	23,96	18,63	14,61	11,43	8,85	6,72	4,95	3,49
0,30	39,91	27,45	20,36	15,50	11,86	9,02	6,74	4,90	3,40	2,20

Продовження табл. Д. 2.1

Ризик постачальника α	Ризик замовника β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,35	35,03	23,56	17,09	12,67	9,43	6,92	4,95	3,39	2,16	1,24
0,40	30,44	19,97	14,11	10,17	7,30	5,12	3,45	2,17	1,23	0,55
0,45	26,13	16,65	11,40	7,92	5,44	3,60	2,23	1,24	0,55	—
0,50	22,09	13,86	8,97	5,95	3,85	2,35	1,28	0,55	—	—
$r = 1,4$										
0,05	41,72	31,39	25,38	21,12	17,830	15,160	12,920	10,980	9,289	7,789
0,10	37,41	27,67	22,02	18,04	14,980	12,510	10,440	8,673	7,141	5,799
0,15	33,56	24,41	19,12	15,42	12,580	10,310	8,417	6,820	5,449	4,264
0,20	30,02	21,46	16,54	13,10	10,500	8,417	6,709	5,280	4,071	3,044
0,25	26,73	18,76	14,20	11,04	8,659	6,779	5,250	3,989	2,940	2,071
0,30	23,65	16,27	12,07	9,19	7,032	5,349	4,001.	2,908	2,020	1,309
0,35	20,76	13,98	10,13	7,52	5,595	4,109	2,940	2,014	1,295	0,735
0,40	18,04	11,84	8,37	6,03	4,331	3,040	2,054	1,302	0,728	—
0,45	15,49	9,87	6,76	4,70	3,229	2,145	1,333	0,734	—	—
0,50	13,10	8,06	5,32	3,54	2,293	1,396	0,759	—	—	—
$r = 1,5$										
0,05	28,03	21,10	17,050	14,190	11,990	10,190	8,680	7,381	6,243	5,235
0,10	25,14	18,60	14,800	12,120	10,070	8,406	7,016	5,830	4,800	3,898
0,15	22,55	16,41	12,850	10,300	8,457	6,927	5,658	4,585	3,664	2,868
0,20	20,18	14,43	11,110	8,808	7,057	5,660	4,512	3,551	2,739	2,048

Продовження табл. Д. 2.1

Ризик постачальника α	Ризик замовника β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,25	17,97	12,61	9,545	7,422	5,823	4,559	3,531	2,684	1,979	1,395
0,30	15,90	10,94	8,116	6,178	4,731	3,600	2,693	1,958	1,362	0,888
0,35	13,96	9,39	6,814	5,061	3,764	2,766	1,980	1,351	0,873	0,495
0,40	12,13	7,96	5,628	4,059	2,915	2,053	1,389	0,875	0,491	—
0,45	10,42	6,64	4,551	3,170	2,182	1,445	0,896.	0,500	—	—
0,50	8,81	5,43	3,590	2,388	1,543	0,939	0,523	—	—	—
$r = 1,6$										
0,05	20,39	15,340	12,400	10,320	8,717	7,411	6,313	5,368	4,541	3,807
0,10	18,28	13,530	10,760	8,818	7,323	6,134	5,104	4,241	3,492	2,836
0,15	16,40	11,930	9,347	7,536	6,152	5,039	4,117	3,336	2,666	2,087
0,20	14,67	10,490	8,086	6,408	5,134	4,118	3,283	2,584	1,994	1,493
0,25	13,07	9,173	6,944	5,400	4,237	3,319	2,571	1,954	1,441	1,018
0,30	11,56	7,956	5,906	4,997	3,444	2,621	1,961	1,427	0,997	0,649
0,35	10,15	6,832	4,960	3,683	2,741	2,015	1,448	0,996	0,635	0,360
0,40	8,83	5,792	4,097	2,956	2,128	1,500	1,012	0,673	—	—
0,45	7,58	4,834	3,318	2,312	1,589	1,052	0,654	—	—	—
0,50	6,41	3,957	2,616	1,739	1,123	0,689	—	—	—	—
$r = 1,7$										
0,05	15,650	11,780	9,519	7,923	6,691	5,689	4,849	4,121	3,486	2,923
0,10	14,030	10,380	8,262	6,769	5,622	4,694	3,918	3,256	2,681	2,178

Продовження табл. Д. 2.1

Ризик постачальника α	Ризик замовника β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,15	12,590	9,160	7,176	5,786	4,723	3,869	3,161	2,562	2,047	1,603
0,20	11,270	8,055	6,028	4,920	3,942	3,163	2,522	1,986	1,532	1,146
0,25	10,030	7,043	5,332	4,148	3,255	2,550	1,976	1,502	1,109	0,787
0,30	8,880	6,111	4,535	3,453	2,645	2,014	1,509	1,101	0,768	0,498
0,35	7,795	5,247	3,809	2,830	2,108	1,555	1,114	0,764	0,488	—
0,40	6,778	4,450	3,149	2,277	1,638	1,152	0,776	0,490	—	—
0,45	5,823	3,717	2,554	1,778	1,220	0,808	0,508	—	—	—
0,50	4,930	3,040	2,009	1,335	0,865	0,540	—	—	—	—
$r = 1,8$										
0,05	12,490	9,399	7,598	6,324	5,341	4,541	3,868	3,289	2,782	2,333
0,10	11,200	8,287	6,595	5,403	4,487	3,747	3,128	2,599	2,141	1,739
0,15	10,050	7,312	5,729	4,619	3,771	3,089	2,524	2,046	1,636	1,282
0,20	8,992	6,430	4,956	3,929	3,148	2,526	2,015	1,586	1,224	0,917
0,25	8,010	5,623	4,258	3,313	2,600	2,036	1,678	1,201	0,889	0,631
0,30	7,089	4,879	3,622	2,758	2,114	1,610	1,210	0,831	0,613	0,396
0,35	6,224	4,191	3,043	2,263	1,688	1,244	0,890	0,610	0,390	—
0,40	5,413	3,557	2,520	1,820	1,309	0,919	0,621	—	—	—
0,45	4,653	2,9Л	2,040	1,420	0,975	0,649	—	—	—	—
0,50	3,936	2,425	1,605	1,068	0,697	—	—	—	—	—

Продовження табл. Д. 2.1

Ризик постачальника α	Ризик замовника β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
$r = 1,9$										
0,05	10,270	7,727	6,246	5,199	4,391	3,733	3,180	2,704	2,2881	1,918
0,10	9,209	6,813	5,422	4,423	3,690	3,081	2,572	2,137	1,760	1,430
0,15	8,263	6,012	4,710	3,798	3,100	2,541	2,076	1,684	1,346	1,054
0,20	7,384	5,287	4,076.	0,231	2,590	2,078	1,657	1,305	1,007	0,757
0,25	6,586	4,625	3,02	2,725	2,138	1,675	1,299	0,992	0,734	0,520
0,30	5,029	4,013	2,979	2,269	1,740	1,329	0,997	0,726	0,504	0,325
0,35	5,119	3,447	2,505	1,856	1,390	1,024	0,733	0,502	—	—
0,40	4,—455	2,928	2,075	1,498	1,077	0,756	0,512	—	—	—
0,45	3,828	2,444	1,677	1,168	0,803	0,538	—	—	—	—
0,50	3,237	1,997	1,321	0,881	0,581	—	—	—	—	—
$r = 2,0$										
0,05	8,638	6,501	5,255	4,374	3,694	3,141	2,676	2,275	1,924	1,614
0,10	7,748	5,733	4,562	3,728	3,104	2,592	2,165	1,799	1,482	1,204
0,15	6,952	5,059	3,963	2,196	2,610	2,138	1,748	1,417	1,133	0,887
0,20	6,221	4,449	3,490	2,710	2,179	1,749	1,395	1,099	0,849	0,639
0,25	5,542	3,892	2,947	2,293	1,799	1,411	1,097	0,936	0,619	0,436
0,30	4,905	3,377	2,508	1,912	1,469	1,119	0,840	0,610	0,424	—
0,35	4,309	2,904	2,111	1,572	1,171	0,860	0,616	0,423	—	—
0,40	3,750	2,465	1,747	1,261	0,905	0,637	0,434	—	—	—

Продовження табл. Д. 2.1

Ризик постачальника α	Ризик замовника β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,45	3,222	2,056	1,411	0,983	0,678	0,459	—	—	—	—
0,50	2,724	1,681	1,113	0,746	—	—	—	—	—	—
$r = 2,5$										
0,05	4,542	3,418	2,763	2,300	1,942	1,652	1,407	1,197	1,013	0,849
0,10	4,075	3,015	2,399	1,967	1,634	1,365	1,140	0,947	0,781	0,634
0,15	3,657	2,662	2,066	1,682	1,374	1,226	0,921	0,747	0,599	0,471
0,20	3,273	2,342	1,806	1,432	1,149	0,924	0,739	0,584	0,451	0,336
0,25	2,918	2,051	1,556	1,212	0,953	0,747	0,579	0,441	0,324	—
0,30	2,587	1,783	1,325	1,011	0,774,	0,590	0,442	0,321	—	—
0,35	2,272	1,531	1,112	0,827	0,616	0,455	—	—	—	—
0,40	1,974	1,298	0,919	0,665	0,483	—	—	—	—	—
0,45	1,696	1,085	0,749	0,530	—	—	—	—	—	—
0.50	1,438	0,895	—	—	—	—	—	—	—	—
$r = 3,0$										
0,05	2,942	2,214	1,790	1,490	1,259	1,070	0,912	0,776	0,656	0,551
0,10	2,640	1,954	1,555	1,275	1,059	0,884	0,739	0,614	0,506	0,412
0,15	2,370	1,725	1,352	1,091	0,892	0,732	0,599	0,487	0,391	0,307
0,20	2,123	1,520	1,174	0,932	0,748	0,601	0,480	0,378	0,291	0,217
0,25	1,894	1,331	1,010	0,785	0,617	0,484	0,375	0,285	—	—

Закінчення табл. Д. 2.1

Ризик постачальника α	Ризик замовника β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,30	1,676	1,155	0,859	0,654	0,502	0,383	—	—	—	—
0,35	1,472	0,992	0,722	0,538	0,404	—	—	—	—	—
0,40	1,280	0,843	0,601	0,440	—	—	—	—	—	—
0,45	1,105	0,712	—	—	—	—	—	—	—	—
0,50	0,945	—	—	—	—	—	—	—	—	—
$r = 3,5$										
0,05	2,126	1,600	1,294	1,077	0,910	0,774	0,660	0,561	0,475	0,398
0,10	1,908	1,412	1,124	0,922	0,736	0,640	0,535	0,445	0,368	0,330
0,15	1,714	1,248	0,979	0,791	0,647	0,531	0,435	0,353	0,283	0,221
0,20	1,537	1,101	0,849	0,674	0,513	0,434	0,346	0,272	0,209	—
0,25	1,370	0,963	0,730	0,568	0,445	0,350	0,271	—	—	—
0,30	1,212	0,835	0,621	0,474	0,365	—	—	—	—	—
0,35	1,065	0,719	0,525	0,394	—	—	—	—	—	—
0,40	0,950	0,616	0,444	—	—	—	—	—	—	—
0,45	0,807	—	—	—	—	—	—	—	—	—

ЗНАЧЕННЯ СЕРЕДНЬОЇ ТРИВАЛОСТІ ВИПРОБУВАНЬ h_1

Таблиця Д. 2.2

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
$r = 1,2$										
0,05	169,3	151,8	136,1	121,8	108,5	95,95	84,21	73,17	62,80	53,070
0,10	126,9	112,3	99,08	87,09	76,12	66,00	56,64	47,99	40,00	32,660
0,15	103,0	89,39	78,61	67,11	57,61	48,96	41,08	33,91	27,40	21,540
0,20	85,75	73,24	62,57	53,18	44,79	37,26	30,49	24,43	19,03	14,280
0,25	72,42	60,82	51,08	42,60	35,13	28,52	22,67	17,53	13,05	9,218
0,30	61,58	50,78	41,84	34,17	27,50	21,69	16,64	12,30	8,63	5,597
0,35	52,46	42,39	34,17	27,23	21,30	16,22	11,90	8,29	5,34	3,041
0,40	44,61	35,23	27,69	21,43	16,18	11,78	8,14	5,22	2,95	1,340
0,45	37,74	29,01	22,13	16,53	11,93	8,18	5,21	2,92	1,32	—
0,50	31,65	23,57	17,33	12,36	8,40	5,29	2,96	1,32	—	—
$r = 1,3$										
0,05	83,96	75,26	67,51	60,39	53,77	47,57	41,750	36,280	31,130	26,310
0,10	63,18	55,69	49,12	43,18	37,74	32,72	28,090	23,790	19,840	16,200
0,15	51,08	44,32	38,48	33,28	28,67	24,28	20,380	16,820	13,590	10,690
0,20	42,52	36,32	31,03	26,37	22,22	18,48	15,130	12,120	9,447	7,094
0,25	35,91	30,17	25,34	21,13	17,43	14,15	11,260	8,701	6,482	4,582
0,30	30,54	25,19	20,75	16,95	13,65	10,77	8,266	6,114	4,293	2,792
0,35	26,02	21,03	16,96	13,52	10,57	8,06	5,919	4,128	2,662	1,528

Продовження табл. Д. 2.2

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,40	22,13	17,48	13,75	10,65	8,04	5,86	4,058	2,600	1,488	0,667
0,45	18,73	14,41	10,99	8,21	5,93	4,07	2,600	1,472	0,658	—
0,50	15,71	11,70	8,61	6,15	4,18	2,66	1,486	0,666	—	—
$r = 1,4$										
0,05	52,26	46,87	42,04	37,60	33,480	29,630	26,000	22,600	19,390	16,390
0,10	39,34	34,68	30,60	26,90	23,510	20,390	17,500	14,830	12,360	10,100
0,,15	31,81	27,61	23,97	20,73	17,800	15,130	12,700	10,490	8,475	6,664
0,20	26,45	22,63	19,34	16,44	13,850	11,520	9,434	7,562	5,896	4,428
0,25	22,39	18,80	15,79	13,17	10,870	8,829	7,021	5,433	4,048	2,865
0,30	19,03	15,70	12,94	10,57	8,513	6,719	5,163	3,821	2,684	1,746
0,35	16,22	13,11	10,58	8,44	6,602	5,033	3,698	2,580	1,681	0,561
0,40	13,61	10,91	8,58	6.65	5,025	3,660	2,542	1,645	0,929	—
0,45	11,69	8,99	6,86	5,13	3,709	2,562	1,638	0,916	—	—
0,50	9,81	7,30	5,38	3,85	2,633	1,644	0,927	—	—	—
$r = 1,5$										
0,05	36,74	32,99	29,600	26,480	23,580	20,850	18,310	15,910	13,660	11,540
0,10	27,70	24,43	21,550	18,940	16,560	14,360	12,330	10,450	8,709	7,114
0,15	22,41	19,44	16,890	14,610	12,540	10,660	8,949	7,389	5,975	4,700
0,20	18,66	15,94	13,620	11,580	9,760	8,124	6,652	5,333	4,159	3,125
0,25	15,77	13,25	11,130	9,286	7,664	6,226	4,952	3,835	2,859	2,024

Продовження табл. Д. 2.2

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,30	13,41	11,07	9,124	7,456	6,008	4,746	3,645	2,699	1,899	1,248
0,35	11,43	9,24	7,461	5,954	4,660	3,553	2,613	1,835	1,191	0,675
0,40	9,73	7,69	6,053	4,691	3,546	2,598	1,809	1,159	0,656	—
0,45	8,24	6,34	4,843	3,629	2,637	1,815	1,153	0,656	—	—
0,50	6,91	5,16	3,813	2,731	1,858	1,172	0,675	—	—	—
$r = 1,6$										
0,05	27,06	25,070	22,490	20,120	17,910	15,850	13,910	12,090	10,380	8,773
0,10	21,05	18,560	16,370	14,400	12,580	10,910	9,370	7,941	6,621	5,409
0,15	17,03	14,780	12,840	11,100	9,533	8,107	6,807	5,621	4,545	3,575
0,20	14,18	12,120	10,360	8,808	7,424	6,178	6,061	4,058	3,169	2,383
0,25	11,98	10,070	8,463	7,062.	5,831	4,741	3,773	2,920	2,177	1,546
0,30	10,20	8,417	6,943	5,676	4,574	3,612	2,775	2,059	1,459	0,955
0,35	8,70	7,035	5,681	4,530	3,549	2,709	2,005	1,406	0,905	0,512
0,40	7,40	5,852	4,608	3,575	2,715	1,989	1,378	0,882	—	—
0,45	6,27	4,829	3,699	2,776	2,009	1,379	0,880	—	—	—
0,50	5,27	3,943	2,909	2,079	1,413	0,901	—	—	—	—
$r = 1,7$										
0,05	22,360	20,050	17,990	16,090	14,330	12,680	11,130	9,673	8,304	7,020
0,10	16,840	14,850	13,100	11,520	10,070	8,734	7,498	6,355	5,300	4,331
0,15	13,620	11,830	10,270	8,885	7,631	6,491	5,450	4,500	3,639	2,865

Продовження табл. Д. 2.2

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,20	11,350	9,699	8,292	7,051	5,942	4,949	4,056	3,255	2,540	1,907
0,25	5,592	8,063	6,778	5,659	4,673	3,799	3,023	2,340	1,749	1,252
0,30	8,168	6,743	5,560	4,545	3,663	2,894	2,230	1,663	2,174	0,734
0,35	6,964	5,635	4,547	3,630	2,850	2,188	1,613	1,124	0,724	—
0,40	5,929	4,691	3,698	2,880	2,183	1,593	1,101	0,708	—	—
0,45	5,029	3,881	2,976	2,227	1,608	1,104	0,715	—	—	—
0,50	4,235	3,160	2,329	1,663	1,135	0,742	—	—	—	—
$r = 1,8$										
0,05	18,550	16,640	14,930	13,350	11,890	10,520	9,237	8,028	6,893	5,827
0,10	13,580	12,320	10,870	9,561	8,359	7,251	6,225	5,278	4,403	3,598
0,15	11,310	9,817	8,530	7,378	6,337	5,389	4,526	3,740	3,027	2,385
0,20	9,423	8,053	6,885	5,857	4,939	4,114	3,373	2,704	2,109	1,588
0,25	7,569	6,698	5,633	4,703	3,884	3,156	2,512	1,950	1,465	1,047
0,30	6,785	5,602	4,619	3,777	3,046	2,412	1,868	1,388	0,976	0,631
0,35	5,783	4,682	3,782	3,026	2,381	1,824	1,340	0,933	0,601	—
0,40	4,928	3,907	3,088	2,398	1,817	1,322	0,915	—	—	—
0,45	4,190	3,232	2,474	1,850	1,135	0,923	—	—	—	—
0,50	3,515	2,626	1,934	1,384	0,955	—	—	—	—	—
$r = 1,9$										
0,05	15,820	14,190	12,730	11,390	10,140	8,977	7,881	6,850	5,881	4,973

Продовження табл. Д. 2.2

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,10	11,520	10,510	9,277	8,159	7,133	6,188	5,314	4,505	3,758	3,071
0,15	9,650	8,379	7,278	6,298	5,409	4,602	3,865	3,197	2,587	2,036
0,20	8,043	6,874	5,880	5,002	4,219	3,515	2,879	2,308	1,804	1,362
0,25	6,804	5,721	4,809	4,016	3,314	2,695	2,150	1,677	1,259	0,896
0,30	5,789	4,782	3,945	3,227	,2,608	2,074	1,602	1,186	0,832	0,534
0,35	4,939	4,002	3,239	2,597	2,039	1,559	1,144	0,796	—	—
0,40	4,221	3,346	2,642	2,050	1,550	1,128	0,785	—	—	—
0,45	3,582	2,760	2,110	1,578	1,141	0,798	—	—	—	—
0,50	3,003	2,240	1,652	1,187	0,833	—	—	—	—	—
$r = 2,0$										
0,05	13,780	12,370	11,090	9,926	8,840	7,823	6,869	5,970	5,126	4,333
0,10	10,390	9,164	8,087	7,112	6,218	5,394	4,634	3,928	3,277	2,680
0,15	8,412	7,303	6,344	5,489	4,718	4,015	3,375	2,789	2,257	1,776
0,20	7,013	5,995	5,129	4,365	3,680	3,065	2,510	2,015	1,578	1,199
0,25	5,932	4,988	4,193	3,502	2,892	2,356	1,887	1,469	1,102	0,779
0,30	5,048	4,171	3,443	2,823	2,289	1,814	1,398	1,033	0,723	—
0,35	4,313	3,500	2,835	2,271	1,781	1,356	0,996	0,695	—	—
0,40	.3,684	2,922	2,305	1,787	1,348	0,985	0,691	—	—	—
0,45	3,124	2,406	1,838	1,376	1,002	0,710	—	—	—	—
0,50	2,617	1,954	1,444	1,045	—	—	—	—	—	—

Продовження табл. Д. 2.2

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
$r = 2,5$										
0,05	8,449	7,580	6,803	6,088	5,424	4,801	4,215	3,665	3,147	2,662
0,10	6,374	5,624	4,964	4,369	3,823	3,319	2,853	2,420	2,019	1,650
0,15	5,170	4,491	3,903	3,377	2,902	2,469	2,075	1,718	1,399	1,114
0,20	4,308	3,684	3,153	2,686	2,271	1,903	1,568	1,265	0,991	0,744
0,25	3,652	3,079	2,601	2,180	1,806	1,470	1,170	0,905	0,670	—
0,30	3,131	2,592	2,140	1,753	1,411	1,113	0,885	0,631	—	—
0,35	2,670	2,162	1,745	1,392	1,090	0,836	—	—	—	—
0,40	2,266	1,793	1,413	1,099	0,845	—	—	—	—	—
0,45	1,917	1,482	1,144	0,881	—	—	—	—	—	—
0,50	1,620	1,228	—	—	—	—	—	—	—	—
$r = 3,0$										
0,05	6,210	5,571	5,000	4,475	3,987	3,530	3,102	2,699	2,321	1,965
0,10	4,696	4,144	3,658	3,218	2,815	2,442	2,098	1,780	1,485	1,222
0,15	3,801	3,303	2,872	2,489	2,144	1,834	1,551	1,290	1,051	0,831
0,20	3,184	2,732	2,348	2,004	1,695	1,415	1,160	0,931	0,722	0,532
0,25	2,714	2,286	1,925	1,609	1,327	1,078	0,856	0,659	—	—
0,30	2,306	1,905	1,572	1,283	1,034	0,820	—	—	—	—
0,35	1,960	1,587	1,282	1,028	0,618	—	—	—	—	—
0,40	1,668	1,326	1,057	0,843	—	—	—	—	—	—

Закінчення табл. Д. 2.2

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
1	2	3	4	5	6	7	8	9	10	11
0,45	1,429	1,122	—	—	—	—	—	—	—	—
0,50	1.238	—	—	—	—	—	—	—	—	
$r = 3,5$										
0,05	4,999	4,487	4,030	3,609	3,217	2,851	2,505	2,1792	1.87	1,583
0,10	3,779	3,334	2,944	2,590	2,266	1,969	1,695	1,446	1,221	1,010
0,15	3,069	2,674	2,334	2,030	1,753	1,497	1,263	1,046	0,847	0,662
0,20	2,589	2.219	1,902	1,620	1,365	1,133	0,931	0,742	0,569	—
0,25	2,187	1,841	1,547	1,289	1,064	0,865	0,686	—	—	—
0,30'	1,852	1,530	1,262	1,035	0,841	—	—	—	—	—
0,35	1,578	1,282	1,045	0,852	—	—	—	—	—	—
0,40	1,359	1,093	0,892	—	—	—	—	—	—	—
0,45	1,188	—	—	—	—	—	—	—	—	—

ЗНАЧЕННЯ ВЕЛИЧИНИ $\ln \bar{A}$

Таблиця Д. 2.3

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
$r = 1,2$										
0,05	2,884	г2,830	2,772	2,712	2,647	2,578	2,504	2,424	2,337	2,242
0,10	2,185	2,136	2,094	2,019	1,954	1,885	1,811	1,731	1,644	1,549
0,15	1,784	1,731	1,674	1,613	1,549	1,480	1,405	1,325	1,238	1,143
0,20	1,497	1,443	1,386	1,325	1,261	1,191	1,118	1,038	0,951	0,855
0,25	1,274	1,220	1Д63	1,102	1,038	0,969	0,894	0,814	0,727	0,632
0,30	1,092	1,038	0,981	0,920	0,855	0,786	0,712	0,632	0,545	0,450
0,35	0,938	0,884	0,826	0,766	0,701	0,632	0,558	0,478	0,391	0,297
0,40	0,804	0,750	0,693	0,632	0,568	0,499	0,424	0,343	0,270	0,614
0,45	0,686	0,632	0,575	0,515	0,450	0,381	0,308	0,226	0,148	—
0Г50	0.581	0,527	0,469	0,409	0,343	0,276	0,197	0,131	—	—
$r = 1,3$										
0,05	2,860	2,804	2,747	2,686	2,621	2,552	2,478	2,398	2,311	2,215
0,10	2,165	2,110	2,053	1,992	1,926	1,858	1,785	1,704	1,617	1,522
0,15	1,759	1,705	1,647	1,587	1,522	1,453	1,376	1,298	1,211	1,116
0,20	1,471	1,417	1,360	1,298	1,234	1,165	1,091	1,011	0,924	0,829
0,25	1,248	1,194	1,137	1,075	1,011	0,941	0,889	0.788	0,701	0,606
0,30	1,065	1,013	0,954	0,893	0,829	0,759	0,685	0,606	0,517	0,426

Продовження табл. Д. 2.3

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,35	0,911	0,857	0,800	0,739	0,674	0,606	0,531	0,452	0,367	0,258
0,40	0,778	0,723	0,667	0,606	0,541	0,472	0,400	0,315	0,235	0,147
0,45	0,660	0,606	0,548	0,486	0,425	0,356	0,270	0,210	0,212	—
0,50	0,554	0,499	0,444	0,384	0,316	0,247	0,188	0,075	—	—
$r = 1,4$										
0,05	2,833	2,778	2,721	2,666	2,596	2,527	2,452	2,372	2,286	2,191
0,10	2,139	2,085	2,028	1,967	1,903	1,834	1,760	1,680	1,593	1,497
0,15	1,734	1,680	1,622	1,562	1,497	1,428	1,354	1,274	1,187	1,092
0,20	1,446	1,392	1,335	1,274	1,210	1,141	1,067	0,987	0,899	0,805
0,25	1,223	1,169	1,112	1,051	0,987	0,918	0,844	0,764	0,674	0,581
0,30	1,044	0,987	0,929	0,869	0,804	0,735	0,659	0,581	0,498	0,392
0,35	0,087	0,832	0,775	0,715	0,648	0,581	0,510	0,427	0,326	0,260
0,40	0,753	0,699	0,640	0,581	0,520	0,449	0,364	0,298	0,224	
0,45	0,634	0,581	0,526	0,465	0,395	0,323	0,269	0,184	—	—
0,50	0,532	0,478	0,418	0,346	0,297	0,240	0,146	—	—	—
$r = 1,5$										
0,05	2,870	2,755	2,698	2,638	2,573	2,504	2,430	2,350	2,263	2,168
0,10	2,116	2,062	2,005	1,944	1,880	1,811	1,737	1,657	1,570	1,474
0,15	1,711	1,657	1,600	1,539	1,474	1,405	1,331	1,251	1,164	1,069
0,20	1,423	1,369	1,312	1,251	1,187	1,118	1,044	0,965	0,878	0,777

Продовження табл. Д. 2.3

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,25	1,200	1,146	1,089	1,028	0,965	0,896	0,819	0,739	0,656	0,552
0,30	1,018	0,965	0,907	0,846	0,779	0,712	0,641	0,562	0,461	0,374
0,35	0,864	0,807	0,750	0,693	0,630	0,561	0,479	0,391	0,333	0,244
0,40	0,729	0,677	0,622	0,561	0,491	0,406	0,357	0,292	0,188	—
0,45	0,615	0,560	0,499	0,428	0,375	0,322	0,250	0,130	—	—
0,50	0,506	0,445	0,389	0,344	0,283	0,210	0,072	—	—	—
$r = 1,6$										
0,05	2,788	2,734	2,677	2,616	2,552	2,483	2,408	2,328	2,241	2,146
0,10	2,095	2,041	1,984	1,923	1,858	1,789	1,715	1,635	1,548	1,453
0,15	1,689	1,635	1,578	1,517	1,453	1,384	1,310	1,230	1,144	1,048
0,20	1,402	1,348	1,290	1,230	1,166	1,097	1,023	0,939	0,853	0,765
0,25	1,178	1,125	1,068	1,007	0,940	0,871	0,800	0,724	0,635	0,523
0,30	0,996	0,940	0,883	0,824	0,763	0,696	0,619	0,523	0,443	0,376
0,35	0,841	0,789	0,735	0,674	0,607	0,526	0,451	0,398	0,320	0,216
0,40	0,712	0,657	0,597	0,528	0,458	0,411	0,353	0,264	—	—
0,45	0,590	0,529	0,463	0,422	0,371	0,308	0,208	—	—	—
0,50	0,478	0,431	0,387	0,334	0,262	0,152	—	—	—	—
$r = 1,7$										
0,05	2,767	2,714	2,657	2,596	2,532	2,463	2,389	2,309	2,222	2,126
0,10	2,075	2,021	1,964	1,903	1,839	1,770	1,696	1,616	1,528	1,432

Продовження табл. Д. 2.3

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,15	1,670	1,616	1,557	1,497	1,432	1,364	1,291	1,211	1,222	1,022
0,20	1,381	1,328	1,272	1,211	1,146	1,075	0,999	0,922	0,841	0,745
0,25	1,159	1,104	1,045	0,984	0,922	0,857	0,786	0,701	0,596	0,503
0,30	0,975	0,922	0,867	0,809	0,744	0,670	0,579	0,505	0,447	0,368
0,35	0,825	0,772	0,713	0,647	0,569	0,506	0,456	0,390	0,298	—
0,40	0,690	0,630	0,551	0,507	0,462	0,407	0,333	0,224	—	—
0,45	0,555	0,507	0,467	0,418	0,358	0,276	0,148	—	—	—
0,50	0,471	0,427	0,375	0,310	0,222	0,074	—	—	—	—
$r = 1,8$										
0,05	2,749	2,695	2,638	2,577	2,513	2,444	2,370	2,290	2,203	2,107
0,10	2,056	2,002	1,945	1,884	1,820	1,751	1,676	1,595	1,509	1,416
0,15	1,650	1,595	1,539	1,479	1,415	1,346	1,272	1,188	1,099	1,011
0,20	1,364	1,310	1,252	1,189	1,122	1,055	0,986	0,910	0,822	0,717
0,25	1,137	1,083	1,028	0,971	0,909	0,840	0,761	0,664	0,570	0,511
0,30	0,960	0,908	0,851	0,789	0,717	0,631	6,563	0,512	0,442	0,354
0,35	0,806	0,749	0,684	0,608	0,557	0,510	0,453	0,371	0,268	—
0,40	0,659	0,592	0,552	0,509	0,457	0,393	0,300	—	—	—
0,45	0,548	0,508	0,462	0,405	0,336	0,228	—	—	—	—
0,50	0,465	0,415	0,356	0,278	0,160	—	—	—	—	—

Продовження табл. Д. 2.3

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
$r = 1,9$										
0,05	2,731	2,677	2,620	2,559	2,495	2,426	2,352	2,272	2,185	2,090
0,10	2,038	1,984	1,927	1,866	1,801	1,732	1,658	1,579	1,493	1,397
0,15	1,632	1,579	1,522	1,463	1,397	1,326	1,247	1,170	1,089	0,999
0,20	1,345	Г,288	1,230	1,170	1,109	1,044	0,973	0,891	0,794	0,660
0,25	1,122	1,071	1,015	0,957	0,890	0,816	0,725	0,632	0,579	0,513
0,30	0,944	0,890	0,829	0,759	0,679	0,617	0,571	0,510	0,430	0,336
0,35	0,781	0,718	0,645	0,605	0,560	0,509	0,439	0,346	—	—
0,40	0,632	0,595	0,553	0,505	0,446	0,368	0,258	—	—	—
0,45	0,547	0,502	0,449	0,385	0,300	0,170	—	—	—	—
0,50	0,452	0,396	0,328	0,236	0,85	—	—	—	—	—
$r = 2,0$										
0,05	2,714	2,660	2,603	2,542	2,477	2,408	2,335	2,255	2,168	2,072
0,10	2,021	1,967	1,909	1,849	1,784	1,716	1,644	1,564	1,474	1,373
0,15	1,617	1,563	1,505	1,442	1,375	1,305	1,235	1,161	1,079	0,983
0,20	1,323	1,270	1,216	1,160	1,098	1,031	0,955	0,864	0,755	0,661
0,25	1,109	1,057	1,000	0,938	0,866	0,782	0,688	0,642	0,583	0,509
0,30	0,925	0,867	0,800	0,723	0,668	0,623	0,572	0,500	0,414	—
0,35	9,750	0,687	0,650	0,608	0,558	0,499	0,417	0,314	—	—
0,40	0,634	0,594	0,549	0,494	0,428	0,334	0,210	—	—	—

Продовження табл. Д. 2.3

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,45	0,541	0,490	0,431	0,306	0,252	0,104	—	—	—	—
0,50	0,435	0,372	0,292	0,176	—	—	—	—	—	—
$r = 2,5$										
0,05	2,640	2,586	2,528	2,468	2,404	2,336	2,263	2,185	2,098	2,003
0,10	1,948	1,891	1,830	1,768	1,704	1,639	1,572	1,498	1,415	1,325
0,15	1,546	1,495	1,441	1,381	1,318	1,245	1,160	1,065	0,916	0,866
0,20	1,261	1,203	1,139	1,067	0,979	0,903	0,862	0,811	0,751	0,684
0,25	1,007	0,935	0,891	0,852	0,811	0,763	0,707	0,627	0,547	—
0,30	0,847	0,809	0,766	0,718	0,662	0,588	0,449	0,401	—	—
0,35	0,728	0,681	0,626	0,563	0,475	0,368	—	—	—	—
0,40	0,597	0,537	0,467	0,366	0,234	—	—	—	—	—
0,45	0,453	0,373	0,268	0,108	—	—	—	—	—	—
0,50	0,290	0,180	—	—	—	—	—	—	—	—
$r = 3,0$										
0,05	2,585	2,531	2,474	2,412	2,346	2,272	2,191	2,107	2,023	1,934
0,10	1,891	1,841	1,787	1,730	1,669	1,599	1,521	1,437	1,329	1,142
0,15	1,491	1,432	1,355	1,291	1,202	1,103	1,060	1,013	0,961	0,907
0,20	1,150	1,088	1,055	1,017	0,979	0,931	0,872	0,808	0,740	0,670
0,25	0,984	0,945	0,902	0,856	0,798	0,726	0,646	0,561	—	—

Закінчення табл. Д. 2.3

Ризик постачальника α	Ризик споживача β									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,30	0,837	0,788	0,732	0,664	0,575	0,475	—	—	—	—
0,35	0,678	0,612	0,536	0,427	0,302	—	—	—	—	—
0,40	0,590	0,413	0,290	0,130	—	—	—	—	—	—
0,45	0,302	0,172	—	—	—	—	—	—	—	—
0,50	0,074	—	—	—	—	—	—	—	—	—
$r = 3,5$										
0,05	2,523	2,464	2,406	2,345	2,287	2,225	2,157	2,084	2,006	1,925
0,10	1,855	1,800	1,742	1,675	1,599	1,516	1,419	1,256	1,211	1,167
0,15	1,397	1,322	1,246	1,214	1,180	1,137	1,088	1,034	0,979	0,922
0,20	1,148	1,112	1,072	1,030	0,976	0,914	0,846	0,776	0,707	—
0,25	0,978	0,931	0,880	0,814	0,736	0,649	0,560	—	—	—
0,30	0,792	0,730	0,655	0,554	0,444	—	—	—	—	—
0,35	0,589	0,504	0,381	0,238	—	—	—	—	—	—
0,40	0,353	0,228	0,042	—	—	—	—	—	—	—
0,45	0,106	—	—	—	—	—	—	—	—	—

ЗНАЧЕННЯ ВЕЛИЧИНИ $\ln\left[\beta(1-\alpha)^{-1}\right]$ ТА ВЕЛИЧИНИ $-\ln\left[(1-\beta)\alpha^{-1}\right]$

Таблиця Д. 2.4

Ризик постачальника α для величини β $-\ln\left[(1-\beta)\alpha^{-1}\right]$ та ризик споживача β для величини $\ln\left[\beta(1-\alpha)^{-1}\right]$	Ризик споживача β для величини $\ln\left[\beta(1-\alpha)^{-1}\right]$ та ризик постачальника α для величини $-\ln\left[(1-\beta)\alpha^{-1}\right]$									
	0,05	0,10	0,15	0,20	0,25	0,30	0,35	0,40	0,45	0,50
0,05	–2,944	–2,251	–1,846	–1,558	–1,335	–1,153	–0,999	–0,865	–0,747	–0,642
0,10	–2,890	–2,197	–1,792	–1,504	–1,281	–1,099	–0,944	–0,811	–0.6S3	–0,588
0,15	–2,813	–2,140	–1,735	–1,447	–1,224	–1,042	–0,887	–0,754	–0,635	–0,531
0,20	–2,772	–2,079	–1,674	–1,385	–1,163	–0,981	–0,827	–0,693	–0,576	–0,470
0,25	–2,708	–2,015	–1,610	–1,322	–1,099	–0,916	–0,762	–0,629	–0,511	–0,406
0,30	–2,639	–1,946	–1,540	–1,253	–1,030	–0,847	–0,693	–0,560	–0,442	–0,337
0,35	–2,565	–1,872	–1,466	–1,179	–0,956	–0,773	–0,619	–0,619	–0,358	–0,262
0,40	–2,485	–1,792	–1,386	–1,099	–0,875	–0,693	–0,539	–0,406	–0,288	–0,182
0,45	–2,398	–1,705	–1,299	–1,012	–0,780	–0,606	–0,451	–0,319	–0,201	–0,095
0,50	–2,303	–1,609	–1,204	–0,916	–0,693	–0,511	–0,357	–0,223	–0,105	–0,000

ЗНАЧЕННЯ ПАРАМЕТРА R В ЗАЛЕЖНОСТІ ВІД ВІДНОШЕННЯ q_1/q_0 ТА ВІРОГІДНІСТЬ ВІДМОВИ q_0, ЗАДАНОЇ В ТУ

$$r=\frac{\ln\left(1-\frac{q_1}{q_0}q_0\right)}{\ln\left(1-q_0\right)}=\frac{\ln\left\{1-\frac{q_1}{q_0}\left[1-\exp\left(-\frac{t_{6p}}{T_o}\right)\right]\right\}}{-\frac{t_{6p}}{T_o}}$$

Таблиця Д. 2.5

$\frac{t_{6p}}{T_o}$	q_0	q_1/q_0									
		1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	2,0
0,001	0,001	1,1001	1,2001	1,3002	1,4003	1,5004	1,6005	1,7006	1,8008	1,9009	2,0010
0,002	0,002	1,1001	1,2003	1,3004	1,4006	1,5008	1,6010	1,7012	1,8015	1,9018	2,0021
0,003	0,003	1,1002	1,2004	1,3006	1,4009	1,5014	1,6015	1,7018	1,8022	1,9026	2,0031
0,004	0,004	1,1002	1,2005	1,3008	1,4012	1,5016	1,6020	1,7024	1,8029	1,9035	2,0041
0,005	0,005	1,1003	1,2006	1,3010	1,4015	1,5019	1,6025	1,7030	1,8037	1,9043	2,0051
0,006	0,006	1,1003	1,2008	1,3012	1,4017	1,5023	1,6029	1,7036	1,8044	1,9052	2,0061
0,007	0,007	1,1004	1,2009	1,3014	1,4020	1,5027	1,6034	1,7042	1,8051	1,9061	0,0071
0,008	0,008	1,1004	1,2010	1,3016	1,4023	1,5031	1,6039	1,7049	1,8059	1,9070	2,0081
0,009	0,009	1,1005	1,2011	1,3018	1,4026	1,5035	1,6044	1,7055	1,8066	1,9078	2,0092
0,010	0,010	1,1006	1,2012	1,3040	1,4029	1,5036	1,6049	1,7060	1,8073	1,9087	2,0102
0,020	0,020	1,1011	1,2025	1,3040	1,4058	1,5077	1,6099	1,7123	1,8148	1,9137	2,0207
0,030	0,030	1,1017	1,2038	1,3061	1,4087	1,5117	1,6150	1,7186	1,8226	1,9172	2,0315
0,040	0,040	1,1023	1,2050	1,3082	1,4118	1,5158	1,6202	1,7252	1,8305	1,9252	2,0426
0,051	0,050	1,1029	1,2064	1,3103	1,4149	1,5200	1,6256	1,7318	1,8387	1,9365	2,0541

Продовження табл. Д. 2.5

$\frac{t_{бр}}{T_0}$	q_0	q_1/q_0									
		1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	2,0
0,061	0,060	1,1034	1,2077	1,3125	1,4180	1,5242	1,6312	1,7389	1,8471	1,9482	2,0660
0,072	0,070	1,1043	1,2091	1,3148	1,4213	1,5287	1,6368	1,7459	1,8558	1,9602	2,0783
0,083	0,080	1,1048	1,2104	1,3171	1,4246	1,5332	1,6427	1,7532	1,8648	1,9731	2,0911
0,094	0,090	1,1054	1,2119	1,3194	1,4280	1,5378	1,6487	1,7608	1,8740	1,9863	2,1043
0,105	0,100	1,1061	1,2133	1,3218	1,4315	1,5425	1,6548	1,7685	1,8836	2,0000	2,1179
0,116	0,110	1,1068	1,2148	1,3243	1,4351	1,5474	1,6612	1,7765	1,8936	2,0120	2,1321
0,127	0,120	1,1074	1,2163	1,3268	1,4388	1,5524	1,6678	1,7848	1,9037	2,0243	2,1469
0,139	0,130	1,1081	1,2179	1,3294	1,4426	1,5576	1,6745	1,7934	1,9142	2,0371	2,1622
0,150	0,140	1,1088	1,2195	1,3320	1,4465	1,5629	1,6815	1,8022	1,9252	2,0504	2,1781
0,162	0,150	1,1096	1,2210	1,3347	1,4505	1,5684	1,6887	1,8113	1,9365	2,0642	2,1947
0,174	0,160	1,1103	1,2228	1,3375	1,4546	1,5741	1,6961	1,8208	1,9483	2,0786	2,2120
0,186	0,170	1,1110	1,2248	1,3404	1,4588	1,5799	1,7038	1,8306	1,9605	2,0935	2,2300
0,198	0,180	1,1118	1,2263	1,3433	1,4631	1,5859	1,7117	1,8407	1,9731	1,1091	2,2487
0,210	0,190	1,1127	1,2281	1,3463	1,4676	1,5921	1,7198	1,8512	1,9863	1,1240	2,2685
0,223	0,200	1,1135	1,2299	1,3494	1,4722	1,5985	1,7284	1,8621	2,0000	2,1423	2,2892
0,287	0,250	1,1179	1,2399	1,3662	1,4975	1,6338	1,7757	1,9236	2,0781	2,2398	2,4095
0,356	0,300	1,1228	1,2513	1,3859	1,5273	1,6762	1,8338	2,0000	2,1774	2,3662	2,5690
0,430	0,350	1,1285	1,2645	1,4090	1,5631	1,7282	1,9058	2,0982	2,3680	2,5387	2,7949
0,510	0,400	1,1351	1,2802	1,4369	1,6072	1,7938	2,0000	2,2306	2,4920	2,4938	3,1507
0,597	0,450	1,1428	1,2989	1,4711	1,6631	1,8800	2,1293	2,4224	2,7779	3,2300	3,8515
0,692	0,500	1,1520	1,3219	1,5146	1,7370	2,0000	2,3219	2,7370	2,3220	4,3219	—

Продовження табл Д.2.5

$\frac{t_{бр}}{T_0}$	q_0	q_1/q_0									
		2,5	3,0	3,5	4,0	4,5	5,0	5,5	6,0	6,5	7,0
0,001	0,001	2,5019	3,0030	3,5044	4,0060	4,5079	5,0101	5,5125	6,0151	6,5180	7,0211
0,002	0,002	2,5038	3,0060	3,5088	4,0121	4,5159	5,0202	5,5250	6,0303	6,5361	7,0425
0,003	0,003	2,5057	3,0091	3,5132	4,0182	4,5239	5,0303	5,5376	6,0456	6,5549	7,0639
0,004	0,004	2,5076	3,0121	3,5177	4,0243	4,5319	5,0406	5,5503	6,0610	6,5728	7,0857
0,005	0,005	2,5095	3,0152	3,5222	4,0304	4,5400	5,0509	5,5631	6,0766	6,5901	7,1076
0,006	0,006	2,5114	3,0182	3,5267	4,0366	4,5462	5,0613	5,5760	6,0923	6,6102	7,1298
0,007	0,007	2,5133	2,0213	3,5312	4,0429	4,5564	5,0718	5,5890	6,1082	6,6292	7,1522
0,008	0,008	2,5152	3,0244	3,5357	4,0491	4,5646	5,0823	5,6021	6,1242	6,6484	7,1748
0,009	0,009	2,5172	3,0275	3,5403	4,0554	4,5730	5,0929	5,6154	6,1408	6,6770	7,1976
0,010	0,010	2,5191	3,0307	3,5449	4,0618	4,5813	5,1036	5,6287	6,1566	6,6872	7,2207
0,020	0,020	2,5389	3,0627	3,5992	4,1273	4,6682	5,2152	5,7682	6,3275	6,9832	7,4655
0,030	0,030	2,5595	3,0963	3,6442	4,1869	4,7613	5,3356	5,9202	6,5153	7,1214	7,7390
0,040	0,040	2,5810	3,1315	3,6946	4,2711	4,8614	4,4663	6,0864	6,7118	7,3761	8,0472
0,051	0,050	2,6039	31684	3,7504	4,3503	4,9693	5,6081	6,2695	6,9536	7,6627	8,3984
0,061	0,060	2,6267	3,2073	3,8096	4,4353	5,0862	5,7644	6,4723	7,2127	7,9886	8,8036
0,072	0,070	2,6510	3,2482	3,8726	4,5267	5,2134	5,9360	6,6988	7,5062	8,3638	9,2785
0,083	0,080	2,6762	3,2913	3,9398	4,6253	5,3523	6,1264	6,9535	7,8426	8,8025	9,8461
0,094	0,090	2,7027	4,3370	4,0116	4,7321	5,5051	6,3390	7,2441	8,2337	9,3253	10,5423
0,105	0,100	2,7304	3,3853	1,0891	4,8484	5,6742	6,5788	7,5788	8,6967	9,9641	11,4272
0,116	0,110	2,7596	3,4365	4,1716	4,9755	5,8626	6,8522	7,9708	9,2575	10,7717	12,6116

Закінчення табл.Д.2.5

$\frac{t_{бр}}{T_o}$	q_0	q_1/q_0									
		1,1	1,2	1,3	1,4	1,5	1,6	1,7	1,8	1,9	2,0
0,127	0,120	2,7902	3,4911	4,2612	5,1154	6,0745	7,1679	8,4392	9,9580	11,8445	14,3957
0,139	0,130	2,8223	3,5494	4,3584	5,2705	6,3153	7,5385	9,0137	10,8725	13,3872	17,2907
0,150	0,140	2,8699	3,6785	4,5806	5,6381	6,9156	8,5301	10,7247	14,1681	22,6982	—
0,162	0,150	2,8920	3,7506	4,7087	5,8596	7,3011	9,2309	12,1607	18,4618	—	—
0,174	0,160	2,9298	3,6288	4,8509	6,1152	7,7721	10,1815	14,6695	—	—	—
0,186	0,170	3,0125	3,9129	5,0100	6,4145	8,3685	11,6028	23,2056	—	—	—
0,193	0,180	3,0578	4,0052	5,1899	6,7725	9,1639	14,2166	—	—	—	—
0,210	0,190	3,1063	4,1063	5,3955	7,2126	10,3188	—	—	—	—	—
0,223	0,200	3,4094	4,8188	7,2283	—	—	—	—	—	—	—

Список літератури

1. Автоматизированные системы управления воздушным движением. Оценка надежности радиоэлектронных средств АС УВД в процессе испытаний и эксплуатации: Отраслевой стандарт ОСТ 54 71006-85.

2. *Адлер Ю.П., Маркова Е.В., Грановский О.В.* Планирование эксперимента при поиске оптимальных условий. — М.: Наука, 1976. — 254 с.

3. *Блейхут Ричард.* Теория и практика кодов, которые контролируют ошибки. — М.: Мир, 1986. — 576 с.

4. *Борзенко А.Е.* IBM PC: Устройство, ремонт, модернизация. — М.: ТОО Фирма «Компьютер Пресс», 1995. — 298 с.

5. *Виноградов Н.А.* Анализ потенциальных характеристик устройств коммутации и управления сетями новых поколений // Зв'язок. — К., 2004. — № 4. — С. 10—17.

6. *Городецкий А.Я., Заборовский В.С.* Информатика. Фрактальные процессы в компьютерных сетях. — СПб.: Изд-во СПбГТУ, 2000. — 102 с.

7. *Дедков В.К., Северцев Н.А.* Основные вопросы эксплуатации сложных систем. — М.: Высш. шк., 1976. — 406 с.

8. *Демьянчук В.С., Дрововозов В.И., Казимирчак В.В.* Техническая эксплуатация автоматизированных систем управления воздушным движением: Учеб. пособие. — К.: КИИГА, 1988. — 80 с.

9. *Дем'янчук В.С.* Експлуатація технічних і програмних засобів митної служби. — К.: КМУЦА, 1998. — 264 с.

10. *Дмитриев В.И.* Прикладная теория информации. — М.: Высш. шк., 1989. — 319 с.

11. *Дрововозов В.И.* Адаптивная структуризация вычислительной сети // Проблеми системного підходу в економіці: Зб. наук. пр. — К.: НАУ, 2004. — Вип. 7. — С. 126—131.

12. *Дрововозов В.И.* Анализ нагрузки на вычислительные сети автоматизированных систем управления воздушным движением // Проблеми інформатизації та управління: Зб. наук. пр. — К.: НАУ, 2005. — Вип. 12. — С. 58—67.

13. *Дрововозов В.И.* Экспериментальная оценка загруженности воздушного пространства // Проблеми інформатизації та управління: Зб. наук. пр. — К.: НАУ, 2002. — Вип. 5. — С. 90—95.

14. *Дружинин Г.В.* Надежность автоматизированных систем. — М.: Энергия, 1977. — 536 с.

15. *Егоров А.Е., Азаров Г.Н., Коваль А.В.* Исследование устройств и систем автоматики методом планирования эксперимента. — Х.: Изд-во при ХГУ, 1986. — 240 с.

16. *Жук Л.О.* Теорія передачі і перетворення інформації в цивільній авіації. — К.: КІЦА, 1988. — 80 с.

17. *Жуков И.А., Стасюк А.И., Дрововозов В.И.* Концепция организации систем управления воздушным движением с использованием параллельных вычислительных структур // Проблеми здобування, збору та обробки даних повітряної розвідки: Зб. наук. пр. — К.: КІ ВПС, 1998. — С. 99—103.

18. *Жуков І.А., Стасюк А.І., Дрововозов В.І.* Методологія синтезу паралельних процесорів для обчислювальних комплексів систем управління повітряним рухом // Літальні апарати та авіаційні двигуни: Зб. наук. пр. — К.: КІ ВПС, 1998. — Вип. 3. — С. 47—53.

19. *Жуков І.А., Гуменюк В.О., Альтман І.Є.* Комп'ютерні мережі та технології: Навч. посібник. — К.: НАУ, 2004. — 276 с.

20. *Жуков И.А.* Классификация архитектур вычислительных систем с параллельным выполнением вычислительных процессов // УСиМ. — 1995. — № 6. — С. 52—56.

21. *Журахівский Ю.П., Півторак В.П.* Теорія інформації та кодування. — К.: Вища шк., 2001. — 255 с.

22. *Каган Б.М., Мкртумян И.В.* Основы эксплуатации ЭВМ: Учеб. пособие для вузов. — М.: Энергоатомиздат, 1988. — 432 с.

23. *Касами Т., Токура Н.* Теория кодирований. — М.: Мир, 1978. — 576 с.

24. *Колесниченко О.В., Шишигин И.В.* Аппаратные средства PC. — 4-е изд., перераб. и доп. — СПб.: BHV — Санкт-Петербург, 2002. — 1024 с.

25. *Кузьмін І.В., Щрус В.О.* Основи теорії інформації та кодування. — К.: Вища шк., 1986. — 237 с.

26. *Куликовский Л.Ф., Мотов В.В.* Теоретические основы информационных процессов. — М.: Высш. шк., 1987. — 247 с.

27. *Марголис А.* Поиск и устранение неисправностей в персональных компьютерах. — К.: Диалектика, 1994. — 368 с.

28. *Мюллер Скотт.* Модернизация и ремонт ПК / Пер. с англ. — 15-е изд.: — М.: Изд. дом «Вильямс», 2005. — 1344 с.

29. Надежность технических систем: Справочник / Под ред. И.А.Ушакова. — М.: Радио и связь, 1985. — 608 с.

30. *Олифер В.Г., Олифер Н.А.* Компьютерные сети. Принципы, технологии, протоколы: Учеб. для вузов. — 2-е изд. — СПб.: Питер, 2003. — 864 с.

31. *Питерсон У., Уеддон Э.* Коды, которые правят ошибки. — М.: Мир, 1976. — 420 с.

32. *Столингс В.* Современные компьютерные сети. — 2-е изд. — СПб.: Питер, 2003. — 783 с.

33. *Хемминг Ричард.* Теория кодирования и теория информации. — М.: Радио и связь, 1983. — 174 с.

34. *Цимбал В.П.* Теорія інформації та кодування. — К.: Вища шк., 1992. — 264 с.

35. Эксплуатация автоматизированых средств и систем обработки информации: Методические указания и контрольные задания/ Сост.: В.С. Демьянчук, С.Ю. Майков, Г.С. Пяткин. — К.: КМУГА, 1997. — 40 с.

Зміст

Навчальне видання

ЖУКОВ Ігор Анатолійович
ДРОВОВОЗОВ Володимир Іванович
МАСЛОВСЬКИЙ Борис Георгійович

Експлуатація комп'ютерних систем та мереж

Навчальний посібник

Художник обкладинки *Т. Зябліцева*
Редактор *А. Бородавко*
Верстка *А. Борисюк*

Administrators

Users

Server Operators

Everyone

Domain Users