

## **Розділ 1.     ІНФОРМАЦІЙНІ ОСНОВИ КОМП'ЮТЕРНОЇ ЕЛЕКТРОНІКИ**

### **1.1. Електроніка та інформація**

#### **1.1.1. Комп'ютерна електроніка.**

##### ***Покоління розвитку комп'ютерної електроніки***

Комп'ютерна електроніка, основана на головних досягненнях електроніки, є домінуючим засобом створення високопродуктивних комп'ютерних систем. Тому вся історія розвитку та становлення комп'ютерної електроніки нерозривно пов'язана з наукою “електроніка”.

У сучасному трактуванні *електроніка* – це наука про формування та управління потоками електронів у пристроях приймання, передавання, оброблення та збереження інформації. Прослідковуючи історію її розвитку, необхідно звернути увагу на те, що ця наука насамперед забезпечує інформаційні потреби людського суспільства. Розвиток продуктивних сил і виробничих відношень тісно пов'язані з розвитком засобів інформатики. Зокрема, історія розвитку засобів інформаційного спілкування між людьми складається з декількох етапів, які в хронологічному порядку можна описати так: 1) жест і міміка; 2) звукова мова; 3) писемність; 4) книгодрукування; 5) **електроніка**. На сьогодні всі ці засоби опрацювання інформації інтенсивно використовує людське суспільство. Важливим є те, що перехід до кожного нового способу передавання інформації розширював можливості інформаційного спілкування між людьми, що закономірно приводило до стрімкого збільшення продуктивних сил суспільства.

Звукова мова дає можливість виражати різні побажання, думки, ідеї, передавати життєвий та виробничий досвід. Вона забезпечує передавання інформації на малі відстані, але цей спосіб має дуже обмежені можливості збереження та передавання інформації у часі. Принцип письмового вираження усної мови став містком для поширення ідей та досвіду через час і простір. Друк надав переваги писемної мови (знання в області науки, техніки, музики, мистецтва) набуттям великої кількості людей різних епох та географій. Електроніка розширила можливості обміну інформацією між людьми не менше, ніж друк, водночас стрімко збільшивши швидкість, об'єми передавання інформації на величезні відстані, співмірні з розмірами сонячної системи.

*Комп'ютерна електроніка* формує електронну базу функціональних пристроїв, розв'язує схемотехнічні завдання з конструювання функціональних

вузлів та організовує взаємодію між ними через програмне забезпечення. У зв'язку з цим розглянемо хронологічну послідовність розвитку комп'ютерних засобів оброблення інформації з використанням компонентів (вузлів) певних типів.

Період запровадження механічних обчислювальних машин (1642–1939 рр.) вважають **нульовим** поколінням розвитку комп'ютерної електроніки. Його засновником був Блез Паскаль (1642 р), котрий винайшов механічний пристрій з ручним приводом, що давав змогу здійснювати операції додавання і віднімання. Тридцять років по тому німецький математик Лейбніц побудував другу механічну машину, яка здійснювала, ще й окрім того, операції множення та ділення. Лише через 150 років Ч. Бебідж розробив аналітичну машину, що містила пам'ять, вичислювальний пристрій, пристрій вводу-виводу на основі механічних вузлів. Вважають, що це перша спроба створити цифровий комп'ютер. Створена машина потребувала програмування на мові *Ассемблер*. Перші програми були створені Адою Августою Ловлейст, котру через це вважають першою програмісткою в світі. В її честь названа сучасна мова програмування *Ada*.

Оскільки на рубежі 19–20 ст. було винайдено радіо (безпроводне передавання сигналів на відстань), то основні засади тогочасної електроніки були використані для створення вичислювальних пристроїв. Це була ера пасивних елементів: провідників, котушок індуктивностей, магнітів, реле, резисторів, конденсаторів. Тому логічним завершенням цього покоління стало розроблення класу автоматичних лічильних машин з використанням електромагнітних реле. Час виконання команд у них становив 6 сек.

**Перше покоління** (1939–1955 рр.) комп'ютерної електроніки характеризується запровадженням *вакуумної електроніки*. Зазначимо, що ця галузь у своєму розвитку від 20-х рр. минулого століття до початку 50-х років досягла значних успіхів: були створені вакуумні прилади з теоретично передбачуваними параметрами, низьким рівнем шумів та високими коефіцієнтами підсилення. Стимулом для впровадження цих компонентів у вичислювальні засоби стала Друга світова війна, оскільки для нових методів ведення війни та новітньої зброї потрібно було оперативно обробляти інформацію (наприклад, біжучі координати). Зокрема, насамперед для британських збройних сил виникла потреба швидкого розшифрування кодованих сигналів з німецьких підводних човнів, генерованих приладом типу ENIGMA. Результатом стало створення британцями електронного комп'ютера (дешифратора) COLOSSUS – машини Тьюринга (1943 р.). Одночасно у США Моушлі зі своїм студентом Екертоном створює ENIAC (Electronic Numerical Integrator and Computer) – електронний цифровий інтегратор і калькулятор, що містив 18 000 ламп, 1500 реле, 20 регістрів,

6 000 багатоканальних перемикачів, важив 30 тонн і споживав 140 кВт електроенергії.

Найбільш важливим результатом у дослідженнях цього покоління стала сконструйована знаменитим математиком Джон-фон-Нейманом машина IAS (Immediate Address Storage – пам'ять з прямою адресацією). Нейман усвідомив, що програма має бути закладена до пам'яті комп'ютера у цифровій формі разом з даними. Крім того, десяткова система в машині ENIAC, де кожний розряд репрезентований десятьма електронними лампами, має бути замінена бінарною арифметикою. Таким чином запропонована фон-Нейманівська обчислювальна машина складалася з п'яти основних частин: пам'яті; арифметико-логічного пристрою (АЛП); пристрою управління та пристрою вводу-виводу. Пам'ять містила 4 096 слів, кожне слово містило 40 бітів. Всередині АЛП розміщувався особливий внутрішній регістр у 40 бітів (акумулятор). Типова команда додавала слово з пам'яті до акумулятора. Однак ця машина не виконувала арифметичні операції з плаваючою крапкою.

Для цього покоління характерне також впровадження комп'ютерної пам'яті на магнітних сердечниках, розробленої в Масачусетському технологічному інституті (США), та серійним випуском у 1953 році комп'ютера IBM-701, який зберігав лідерство впродовж десяти років.

Для *другого покоління* (1955–1965 рр.) характерним є широкомасштабне впровадження транзисторів (ера напівпровідникових приладів), здатних виконувати всі функції електронної лампи. Ці прилади, як виявилось, мають малі габарити, більш високу надійність та більший термін служби, їх можна експлуатувати у більш жорстких умовах, вони економні у споживанні.

З використанням транзистора, як елемента з ключовими властивостями, можливою стала реалізація багатьох функціональних вузлів комп'ютера та створення елементів пам'яті. Перший комп'ютер на транзисторах (TX-0) був створений у лабораторії Масачусетського технологічного інституту (МТІ) (США) та містив слова зі 16 бітів. Компанія IBM побудувала транзисторну версію IBM-709-7090, а пізніше IBM-7094. Час циклу становив 2 мкс, а пам'ять складалася з 32К по 16 бітів. Комп'ютери цієї серії широко використовували для наукових розрахунків у 60-х роках минулого століття.

У 1964 році компанія CDC випустила комп'ютер 6600, який працював майже на порядок швидше, ніж IBM-7094. Він користувався великою популярністю, оскільки міг виконувати відносно складні розрахунки. Секрет високої швидкості полягав у тому, що всередині центрального процесора знаходилася машина з високим ступенем паралелізму. Вона містила декілька функціональних пристроїв для додавання, множення та ділення з можливістю одночасної роботи. Для комп'ютера цього класу було необхідне високоякісне програмне забезпечення. Розробник – Сеймур Крей, легендарна особа, котрий став відомий у світі

як розробник найбільших суперкомп'ютерів, серед яких: CDC-6600, CDC-7600, Cray-1, Cray-2 та інші.

У 60-тих роках, на початках 70-х відбувалися нові якісні зміни в напівпровідниковій інтегральній електроніці, генерування нових ідей, створення нових напівпровідникових приладів та пристроїв. Особливо важливим було стрімке зростання темпів упровадження нових винаходів та ідей, різке скорочення термінів від наукових відкриттів до їхнього освоєння у виробництві.

Для прикладу, реалізація принципу, на якому основана фотографія потребувала більше ста років (1727–1839 рр.), телефон – більше піввіку (1820–1839 рр.), радіо – лише 25 років, електронне телебачення – 14 років (1922–1936 рр.), електронно-обчислювальна машина – 5 років (1948–1953 рр.), транзистор – 5 років (1948–1953 рр.), інтегральні мікросхеми (ІМС) – 3 роки (1959–1962 рр.).

Характеризуючи бурхливі успіхи другого покоління комп'ютерної електроніки, зазначимо, що реальність створення складних електронних пристроїв та систем, які містять тисячі елементів, спричинила певні протиріччя у розвитку електроніки, насамперед у технології та експлуатації електронних виробів. Зокрема, розроблювані в кінці 1950-х років, комп'ютери містили в середньому близько 100 000 діодів та 25 000 транзисторів. Тому за такої кількості виводів гостро виникла проблема надійності міжз'єднань.

Розв'язання проблеми міжз'єднань привело до створення ІМС. В електроніці виник новий етап – *мікроелектроніка*, що дало можливість розв'язати завдання комплексної мікромініатюризації. Результатом стало зменшення габаритів апаратури, маси, енергоспоживання, матеріалоємності, вартості, збільшення числа виконуваних функцій. Отже ІМС стали елементною базою для створення високоефективних енергоефективних комп'ютерних систем.

Тому *третє покоління* (1965–1980 рр.) комп'ютерної електроніки ґрунтується на основі ВІС ІМС. Успіхи цього покоління нерозривно пов'язані з американським інженером-дослідником Робертом Нойсом, котрого сьогодні вважають автором головних винаходів віку інформаційних технологій. Саме його відкриття зробили комп'ютери більш дешевими і продуктивнішими.

У 1956–57 роках Р. Нойс працював в Shockley Semiconductor Laboratory під керівництвом винахідника транзистора Вільяма Шоклі, а потім разом з сімома колегами заснував одну з перших фірм з виробництва кремнієвих напівпровідників – Fairchild Semiconductor. У липні 1959 року Нойс подав заявку на патент №2981877 “Напівпровідниковий пристрій і струмопровідна структура” — тип інтегральної мікросхеми. За цей винахід (зроблений одночасно з Джеком Кілбі), що по-справжньому перетворив світ, Нойс отримав відзнаки від трьох президентів Сполучених Штатів та велику кількість всіляких нагород.

У 1968 році Нойс і його давній колега Гордон Мур заснували корпорацію Intel (integrated electronics). Через два роки вони створили першу інтегральну мікросхему оперативної пам'яті (DRAM), доступну комерційно.

Розроблення першого МП – це спроба створити універсальну логічну ВІС, налаштовану на виконання конкретної функції засобами програмування після її виготовлення. На таку ІМС спочатку планували лише управляючі функції, однак потім МП почали використовуватися як елементну базу цифрових **в**ичислювальних машин (ЦВМ) четвертого та наступних поколінь. Впровадження МП викликало необхідність розроблення спектра універсальних логічних ВІС, що обслуговують МП: контролери переривань і прямого доступу в пам'ять (ПДП), шинні формувачі, порти вводу/виводу та ін.

Перший МП був розроблений фірмою Intel та промислово вироблений у 1971 році на основі *p-МОП*-технології (i4004). В 1972–73 роках тією ж фірмою були продюзовані моделі i4040 та i8008. Ці МП відносять до до т.зв. *першого покоління*; вони мали дуже обмежені функціональні можливості і дуже швидко були витіснені *другим поколінням*, яке було реалізовано на *n-МОН*-технології, що дало змогу підняти тактову частоту приблизно на порядок, щодо МП першого покоління. Крім того, завдяки впровадженню прогресивних технологій ІМС виникла змога підвищити ступінь інтеграції транзисторів на кристалі, а отже, збільшити складність схеми.

МП другого покоління, найпоширенішим з яких був i8080 (1974 р.), відрізнялися достатньо розвинутою системою команд, наявністю підсистем переривання, прямого доступу до пам'яті, значною кількістю допоміжних ВІС, що забезпечують управління пам'яттю, паралельним і послідовним обміном зі зовнішніми пристроями, реалізацією векторних переривань. Багато ідей, закладених в архітектуру систем на базі 8-розрядного МП i8080, незмінними використовують і в сучасних потужних МП.

Постійне намагання збільшити швидкодію ЕВМ спонукало розробників МП застосувати біполярну інтегральну технологію, насамперед транзисторно-транзисторну логіку (ТТЛ), результатом чого став випуск процесорів *третього покоління*. Звичайно, їхня архітектура суттєво змінилася. Однак ТТЛ у складі ІС мали на той час на порядок вищу швидкодію (щодо *n-МОН*) і, отже, більше споживання потужності. Технологічні можливості не давали змоги використовувати активні способи відведення тепла від кристала, тому використовували єдиний спосіб збереження працездатності чипа в цих умовах – знижували його ступінь інтеграції.

Перший з випущених МП третього покоління – i3000 був дворозрядним! Очевидно, збереження у цьому випадку традиційної архітектури, характерної для МП другого покоління, не привело б до збільшення продуктивності системи, незважаючи на те, що тактова частота зростала майже на порядок. Розв'язок цієї

проблеми зумовив значні структурні зміни в МП третього покоління порівняно з другим: МП виготовляли у вигляді секцій з засобами міжрозрядних зв'язків, що давало змогу об'єднувати в одну систему довільну кількість секцій для досягнення заданої розрядності; пристрій управління був винесений на окремий кристал; за рахунок резерву зовнішніх виводів передбачені окремі шини адреси, введення і виведення даних; кристали управління становили собою управляючий автомат з програмованою логікою, що давало можливість достатньо легко реалізовувати практично довільну систему команд на фіксованій структурі операційного пристрою.

Отже, розробники систем на базі МП третього покоління отримали додаткові ступені свободи: можливість вибрати довільну розрядність процесора та самостійно реалізувати практично довільну систему команд, оптимізовану для розв'язування задач конкретного класу. Оскільки МП у такій архітектурі розміщували на декількох кристалах ВІС, то процесори стали називати *багатокристалними*, на відміну від однокристалних другого покоління.

Характерною ознакою для третього покоління комп'ютерної електроніки було також упровадження мультипрограмування, за якого у пам'яті можливо було записати декілька програм: допоки одна програма чекала, поки закінчиться процес вводу-виводу, інша запускалася до виконання.

**Четверте покоління** (1980–поч. 90-х рр.). Розвиток комп'ютерної електроніки ознаменувався масштабним упровадженням НВІС-мікросхем. Почалася ера персональних комп'ютерів. Перші ПК – у вигляді комплектів, що включали друковану плату, МП Intel 8080, 8-дюймовий дисковід. Програмного забезпечення не було. Операційні системи у цьому поколінні пройшли шлях від MS-DOS до Windows різних версій.

Продовжилися роботи зі створення багатокристалних МП у яких було передбачене збільшення розрядності секцій. Широко поширеними стали комбіновані технології (наприклад, І<sup>2</sup>Л+ ТТЛШ). Паралельно інтенсивно розвивалася архітектура однокристалних МП. Найбільш характерним зразком яких можна вважати сімейство x86 фірми Intel. Розвиток цього напрямку характеризується стрімким зростанням продуктивності процесорів, зумовлених збільшенням розрядності, тактової частоти, реалізацією паралелізму на всіх рівнях роботи процесора та реалізацією інших характерних рішень, властивих “великим” ЕВМ.

У 1983 році дохід корпорації Intel вперше склав мільярд доларів. На початку нашого століття мікропроцесорами “Intel” було обладнано 80% комп'ютерів у самих різних країнах світу.

Характерною ознакою цього покоління стала поява на ринку ПК фірми Apple, а також використання процесорів різних фірм. Почалося виробництво “клонів”.

В 1990-х роках були розроблені суперскалярні процесори, які значно підвищили продуктивність комп'ютерів та можливості графічних інтерфейсів.

**П'яте покоління** (середина 1990-х р. – по сьогодні) є результатом бурхливого розвитку мікроелектроніки, впровадження новітніх технологій у виробництві, а також використання архітектур нового типу.

Попередні етапи розвитку мікроелектроніки, на основі яких розвивалася комп'ютерна електроніка, характеризуються збільшенням ступеня інтеграції ІМС (кількості транзисторів на чипі та зменшенням розмірів їхніх активних елементів). Прогнози четвертого покоління передбачали створення ІМС з кількістю елементів на чипі, рівному  $10^6$ , до 2000 року. Однак унаслідок впровадження новітніх літографічних процесів та використання нових матеріалів така мета була реалізована вже до середини 1990-х років. Вважають, що з цього часу почалася ера **УВІС**– ультравеликих ІС з кількістю елементів на кристалі більше мільйона. Це різко підвищило експлуатаційні параметри всіх мікросхем, зокрема тактову частоту мікропроцесорів, об'єм електронної пам'яті та її швидкодію, про що буде сказано далі.

Для п'ятого покоління характерним є впровадження нових архітектур з розпаралелюванням обчислень, особливостями набору регістрів, формату команд і даних (CISC, RISC, VLIW), а також багатопроцесорних систем (SMP) за принципом взаємодії з пам'яттю.

Зростаючі можливості технології почали використовувати не тільки для збільшення продуктивності процесора, але й для розміщення на кристалі поряд з простим процесором, тих пристроїв, які в традиційній архітектурі розміщувалися на платі поряд з МП у вигляді окремих ВІС: тактовий генератор, ПЗП, ОЗП, порти паралельного і послідовного обміну, контролер переривань, таймери та ін.

### **1.1.2. Розвиток мікроелектроніки та її вплив на становлення елементної бази комп'ютерної електроніки**

Інтегральна технологія (ІТ) за перші 20–30 років свого розвитку досягнула таких темпів росту характеристик якості, яких не знала ні одна область людської діяльності. Дійсно, розглянемо динаміку змін основних параметрів ІТ за перші 20 років її розвитку (1960–1980 рр.): 1) ступінь інтеграції  $N$  зріс на 5–6 порядків; 2) площа транзистора  $S$  зменшилася на 3 порядки; 3) робоча частота  $f$  збільшилася на 1–3 порядки; 4) фактори добротності:  $f \times N$  збільшилася на 5–7 порядків;  $P \times t$  збільшився на 4 порядки, де  $t$  – затримка на елементі,  $P$  – потужність, що розсіюється елементом; 5) надійність збільшилася на 4–8 порядків; 6) продуктивність технології (у транзисторах збільшилася на 4–6 порядків; 7) ціна на транзистор у складі ІС зменшилася на 2–4 порядки.

Американські дослідники підраховали, якщо б авіапромисловість у ті ж роки мала аналогічні темпи розвитку показників якості (вартість- швидкість – розхід палива = вартість – швидкодія – розсіювана потужність), то Боїнг-767 коштував б всього 500\$, земну кулю облітав за 20 хв і витрачав на цей політ 10 л палива.

Успіхи ІТ в області елементної бази дали змогу досягати кристалом все більш високих рівнів ЕВМ: спочатку – логічні елементи, потім операційні елементи (реєстри, лічильники, дешифратори та ін.), операційні пристрої. З цієї точки зору цікаво поглянути на співвідношення ІС логіки і пам'яті у процесі еволюції ІС–СІС–ВІС–НВІС. Перші ІС (ступінь інтеграції  $N \sim 10^1$ ) були виключно логічними елементами. При досягненні  $N$  близько  $10^2$  стали виникати, разом з операційними елементами, перші елементи пам'яті об'ємом у 16–24–128 бітів.

З огляду на подальше зростання ступеня інтеграції пам'ять почала швидко випереджати “логіку”, оскільки за всіма параметрами мала перед логічними схемами переваги. Дійсно структура накопичувача ЗП суттєво регулярна (повторюваність елементів і зв'язків за двома координатами), зв'язність (кількість зовнішніх виведень схеми) її зростає пропорційно логарифму об'єму пам'яті у двічі. Насамкінець, пам'ять потрібна всім і “чим більше, тим краще”, особливо, якщо майже за однакову ціну.

Водночас для ІС-логіки на рівні  $N \sim 10^3$  на кристалі можна вже розмішувати пристрій ЦВМ (наприклад, АЛП, ЦУП), але схеми логіки суттєво нерегулярні, їхня зв'язність зростає приблизно пропорційно  $N$ .

*Геометричний скейлінг мікроелектронних структур. Закон Мура.* Аналіз емпіричних тенденцій перших п'яти років розвитку інтегральної технології дав змогу Гордону Муру сформулювати в 1965 р. деяку закономірність, що згодом була названа *законом Мура*.

Початкове формулювання закону було таким: *кількість транзисторів на чипі (кристал ІМС) подвоюється кожні 12 місяців*. Незважаючи на довільність формування та обґрунтування, закон Мура має певну фізичну основу. Зокрема, його фізичний зміст містить ідею можливості розмірного скейлінгу – масштабного зменшення геометричних розмірів мікроелектронних компонентів зі збереженням електричних і покращенням функціональних характеристик окремих приладів і всієї схеми загалом.

Отож саме можливість геометричного скейлінгу мікроелектронних структур стала технологічною й економічною основою, що забезпечила переможний рух закону Мура впродовж вже більше півстоліття та зумовила зростання ступеня інтеграції схем мікропроцесорів (див.: рис.1.1).



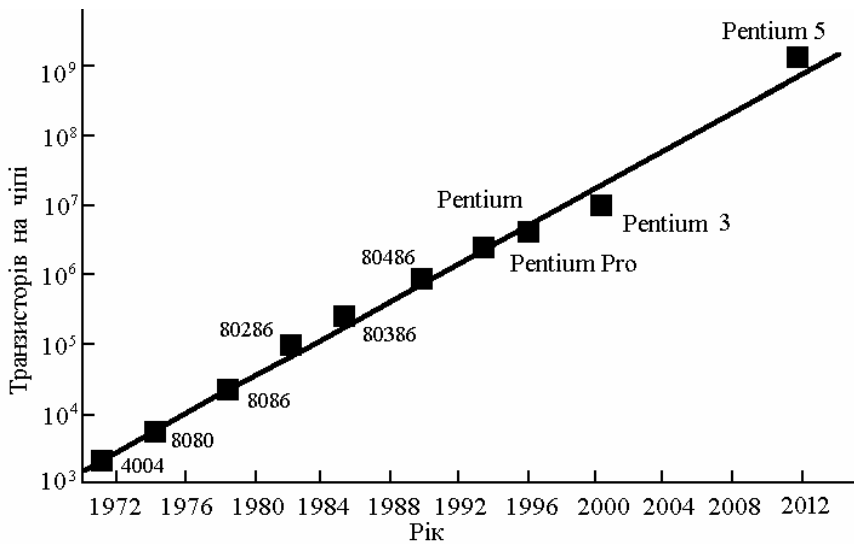


Рис. 1.1. Залежність ступеня інтеграції чіпа від часу

Збільшення ступеня інтеграції чіпа досягається головню за рахунок зменшення технологічної норми та, деякою мірою, за рахунок зростання площі чіпа.

Головним параметром технології є мінімальна технологічна (топологічна, проектна) норма. Насправді під цим параметром розуміють характеристичний розмір для цієї технології. Довжина затвора МОН-транзистора у 1,5 ~ 2 рази менша технологічної норми (див.: рис. 1.2). Довжина каналу зазвичай ще менша від довжини затвора.

Зміна технологічної норми відбувається за поколіннями, з приблизно однаковим масштабним множником  $L \rightarrow L/\sqrt{2} \sim 0,7L$ .

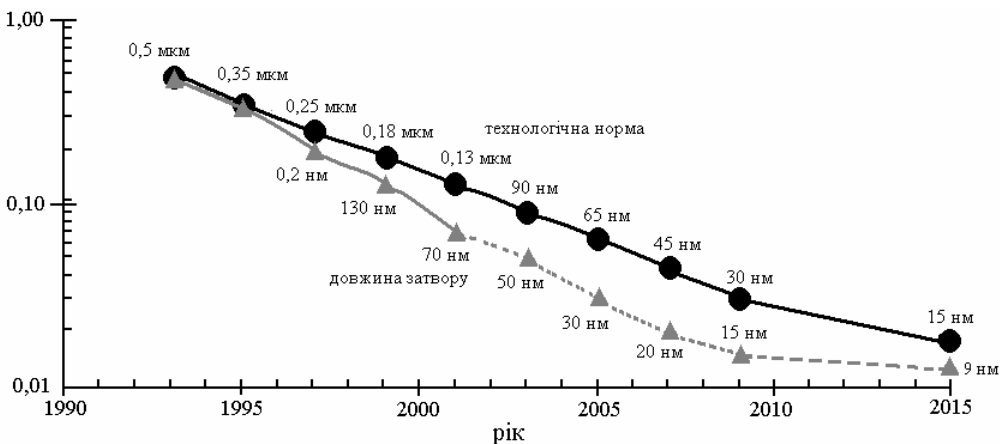


Рис. 1.2. Технологічна норма і довжина затвора для технологій різних поколінь

Перехід до чергового покоління рівнозначний зростанню ступеня інтеграції приблизно у два рази. Очікується, що “кінець ери закону Мура”, тобто коли технологічна норма сучасної кремнієвої технології досягне свого мінімуму  $\sim 5\text{--}10\text{ нм}$ , визначатиметься можливостями літографічних процесів і фундаментальними фізичними обмеженнями (див.: рис. 1.3).

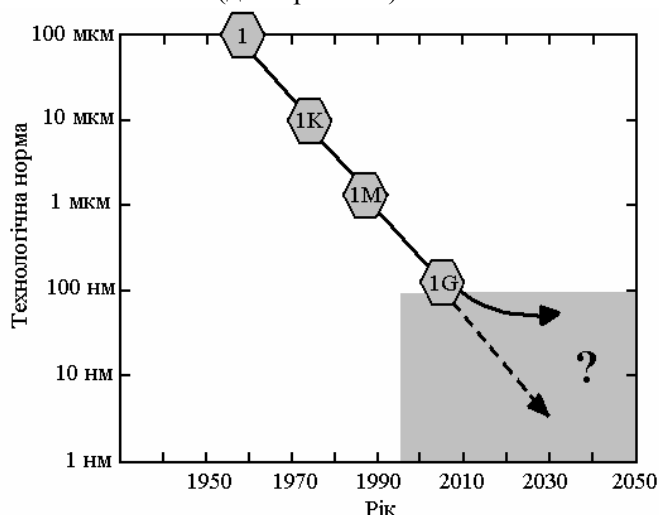


Рис. 1.3. Технологічна норма як функція часу

Усі головні геометричні параметри технології зменшуються разом з технологічною нормою. Це насамперед довжина, ширина каналу транзистора, товщина підзатворного окислу. Залежність геометричного розміру від норми має переважно пропорційний характер.

Однією з найважливіших характеристик мікропроцесора є його швидкодія. Максимальна швидкодія ІС залежить від системної тактової частоти. В останні десятиріччя тактова частота мікропроцесорів збільшувалася у середньому у два рази впродовж року (див.: рис. 1.4).

Тактова частота залежить від швидкості перемикання окремих цифрових елементів, і насамкінець дуже залежить від геометричного фактора  $a$ , отже, від технологічної норми. Встановлено, що тактова частота процесора приблизно подвоюється в кожному наступному поколінні.

Геометричний ресурс для дальшого збільшення тактової частоти практично вичерпаний, оскільки наявне фундаментальне обмеження, зумовлене скінченністю швидкості поширення електричного сигналу.

Основними проблемами при збільшенні ступеня інтеграції та зменшення розмірів активних областей приладів є зростання струмів відтоку та проблема тепловиділення. Ріст ступеня інтеграції та тактової частоти зумовив те, що потужність теплового потоку від поверхні мікропроцесора сучасного ком-

п'ютера загрозливо зростає. Наприклад, якщо перший процесор *Intel 486* давав потік тепла  $5\text{--}8\text{ Вт/см}^2$ , то *Intel Pentium 4*— $30\text{ Вт/см}^2$ .

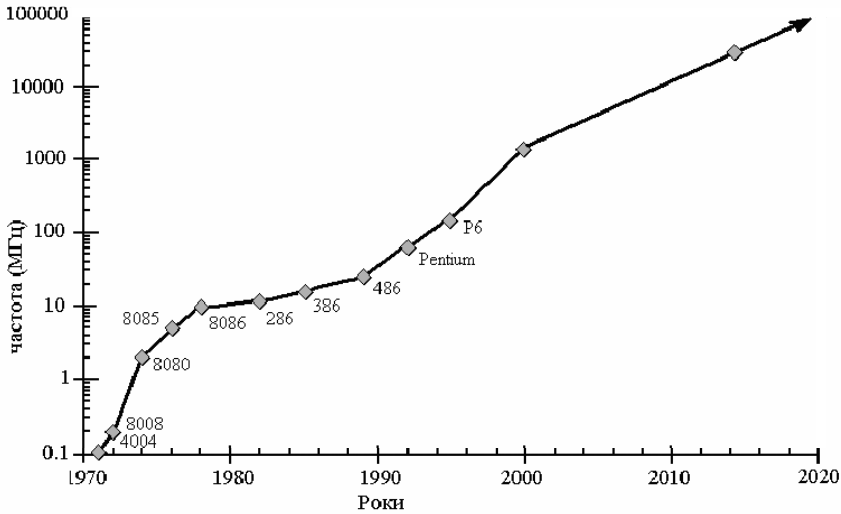


Рис. 1.4. Залежність тактової частоти від року випуску

Фундаментальним параметром, від якого залежить енергоспоживання, є енергія перемикання  $C_{0t}U_{DD}^2$ . Очевидним шляхом розв'язання проблеми мінімізації  $E_S$  є: а) зниження повної ємності, що досягається за рахунок зменшення розмірів елемента; б) зменшення напруги живлення  $U_{DD}$ . У результаті енергія перемикання неперервно зменшується. Наприклад, якщо в 1995 р. енергія перемикання в одиницях  $kT$  становила  $\sim 5 \cdot 10^5$ , то у 2015 році – значно менша –  $\sim 10^3$ .

Разом з тим зменшення розмірів елементів зумовлює підвищення ступеня інтеграції, а потужність потоку тепла лише збільшується. Однак температура мікросхеми під час функціонування має залишатися постійною, тому тепло потрібно відводити. Це є однією з найбільш важливих технічних проблем сучасної мікроелектроніки.

У сьогоденні, з метою створення надшвидкісних комп'ютерів людство почало використовувати кардинально нові принципи оброблення інформації та новітні матеріали, чим увійшло у *шосте покоління* розвитку елементної бази комп'ютерної електроніки. Зазначимо, що у 20 ст. прогрес у підвищенні продуктивності комп'ютерних систем був досягнутий головню за рахунок удосконалення мікроелектронної бази інформатики, підвищення ступеня інтеграції та швидкодії інтегральних схем, використання паралельного та аналогового оброблення зображень, зі спеціалізацією та нарощуванням кількості паралельно

працюючих процесорів. Але створення надшвидкісних комп'ютерів може бути реалізоване лише з використанням нових ідей та принципів, що суттєво модифікують елементну базу. Зважаючи на сказане, сьогодні сформовані декілька напрямків, що ґрунтуються на використанні кардинально нової елементної бази, збудованої на квантових ефектах у новітніх структурах, квантовій інформатиці та квантових обчисленнях. Отже, людство увійшло у *шосте покоління* розвитку елементної бази комп'ютерної електроніки.

До перспективних напрямків розвитку елементної бази цього покоління потрібно віднести створення: одноелектронної елементної бази інформатики (одноелектронний транзистор, одноелектронні логічні схеми, одноелектронні клітинні автомати, одноелектронні елементи і матриці пам'яті); спінтронних елементів та процесорів на їхній основі; молекулярної, вуглецевої та надпровідної елементних баз.

Надпровідні кубіти на переходах Джозефсона вважають на сьогодні найбільш вдалою елементною базою для реалізації квантових процесорів. З 2011 року успішно працює обчислювальна система “D-Wave One” з 128-кубітним квантовим процесором “Rainier”. У 2012 році компанія “D-Wave” успішно випробувала чергову версію надпровідного квантового процесора “Versuvius”, який включає вже 512 кубітів.

Основним недоліком наведеної вище елементної бази є її здатність функціонувати лише при наднизьких температурах ( $\sim 0,1\text{K}$  і нижче), що суттєво стримує можливість її широкого впровадження.

Отже, є всі підстави вважати, що людство вступило в еру квантових комп'ютерів, хоч і перебуває лише на самому її початку.

**Висновки.** Сучасна фізика (особливо квантова) та напівпровідникова електроніка, продукуючи все нові та нові феномени, створюють сприятливі умови для розроблення елементів інформаційних систем, які функціонують на кардинально нових принципах та можуть мати фантастичні експлуатаційні параметри. З цієї причини у довільний момент часу революційні відкриття у цих галузях науки можуть порушити відносно еволюційний рух у розробленні функціональних елементів, що склався сьогодні, а як наслідок – докорінно змінити елементну базу. Свідченням цього є вся історія розвитку комп'ютерної техніки, а особливо період переходу комп'ютерної техніки на елементну базу напівпровідникових ІМС. Тому автори не намагаються сьогодні передбачати конкретний тип функціональних елементів інформаційних систем не лише на майбутні десятиріччя, але й на найближчі десять років. Закон Мура, що на сьогодні є основним “дороговказом” у напівпровідниковій електроніці і має прогноз свого справдження до 2030 року, буде вичерпаний, очевидно, значно раніше. Наноелектроніку, яка скоріш за все змінить сучасну напівпровідникову мікроелектроніку, будуть описувати “своїм законом Мура”, що матиме, звичай-

но, складніший характер. Загалом, не роблячи прогнозів на домінуюче використання будь-якого перспективного типу функціональних елементів, можна передбачати широке впровадження квантових алгоритмів обчислень та елементів на базі напівпровідникових нанорозмірних структур з надвеликою швидкістю та дуже низьким рівнем енергоспоживання.

## 1.2. Інформація. Кількість інформації

Прогрес людства неминуче веде до збільшення загального об'єму інформації, яким воно користується, причому зростання його відбувається значно швидше, ніж населення земної кулі та його потреби.

Поняття *інформації* є базовим для всіх інформаційних технологій. Будь-яка діяльність людини по суті становить собою процес збирання й опрацювання інформації, прийняття на її основі рішень та їхнього виконання. Спілкування, обмін інформацією властиві всім живим істотам, але особливо людині. З впровадженням сучасних комп'ютерних вичислювальних систем інформація стала одним з найважливіших ресурсів науково-технічного прогресу. Її акумулювання та оброблення з певних позицій дають нові відомості, приводить до нового знання.

Слово *інформація* походить від латинського *informatio*, що в перекладі означає *відомість, роз'яснення, ознайомлення*. В інформатиці використовують таке визначення: *інформація – це відомості, що передаються джерелом отримувачу (приймачу)*.

Інформація завжди пов'язана з матеріальним носієм та має деяке представлення. Вона передається від джерела до приймача в матеріально-енергетичній формі, у формі сигналів, що поширюються у певному середовищі. Інформацію, яка представлена у певній закінченій (системній) формі та передається з одного місця в інше, називають *повідомленням*. *Джерело інформації* – це суб'єкт або об'єкт, що породжує інформацію та подає її у вигляді повідомлення. Його подають як сигнали і дані. Сигнали використовують для передавання інформації у просторі між джерелом і приймачем, а дані – для збереження. *Приймач інформації* – це суб'єкт або об'єкт, що приймає повідомлення. Сукупність технічних засобів, використовуваних для передавання повідомлень від джерела до приймача, називають *системою зв'язку*. *Канал зв'язку* – це сукупність технічних пристроїв або фізичне середовище, що забезпечує передавання сигналів від джерела до приймача. Кодуючий пристрій призначений для перетворення інформації вихідного повідомлення від джерела до вигляду, необхідного для передавання інформації. Декодуєчий пристрій призначений для перетворення отриманого повідомлення у вихідний, придатний для його

наступного оброблення. Узагальнена схема передавання інформації технічною системою зображена на рисунку 1.5.

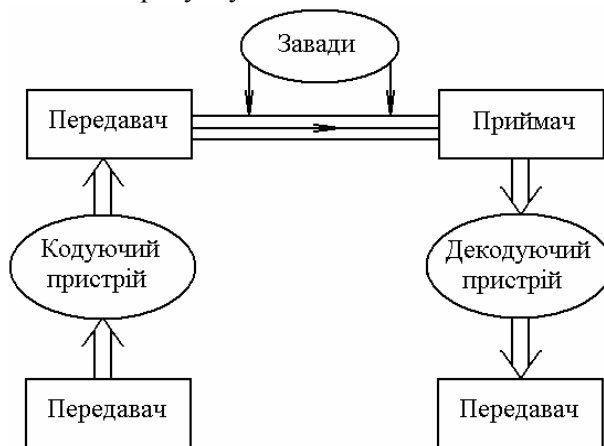


Рис. 1.5. Схематичне зображення процесу передавання інформації

Перше чітке визначення інформації дав американський вчений К. Шеннон у 1948 році на основі ймовірно-статистичного підходу. Він визначив її як *міру зменшення невизначеності*, тобто *відбирання необхідних елементів з деякої їхньої сукупності*. Тобто у цьому розумінні інформація – *це відомості, що знімають невизначеність, наявну до їхнього отримання*. Згідно з визначенням, оснований на комбінаториці, зробленим англійським нейрофізіологом У. Ешбі (1956 р.), інформація детермінується не як ліквідація невизначеності, а як зняття одноманітності, тотожності. Мірою кількості інформації тут служить *ступінь різноманітності елементів системи або відомостей про неї*. Одиницею вимірювання кількості інформації є *біт*, який відповідає вибору одного з двох рівноможливих станів або двох рівноможливих ймовірностей.

Інформація має властивість *адитивності*: загальна кількість інформації, необхідної для розв'язання двох задач, рівна сумі окремих інформацій. Тому, якщо задано число рівноймовірних розв'язків задачі, то інформація пропорційна натуральному логарифму цього числа.

Щоб оцінити кількість інформації використовують ймовірнісний підхід, оскільки за Шенноном кожне повідомлення характеризується ймовірністю появи. Ймовірність певних типів повідомлень встановлюють на основі статистичного аналізу.

Нехай на вхід системи передавання інформації (СПІ) від джерела інформації надходить сукупність повідомлень, вибраних з ансамблю повідомлень (див.: рис. 1.6.).

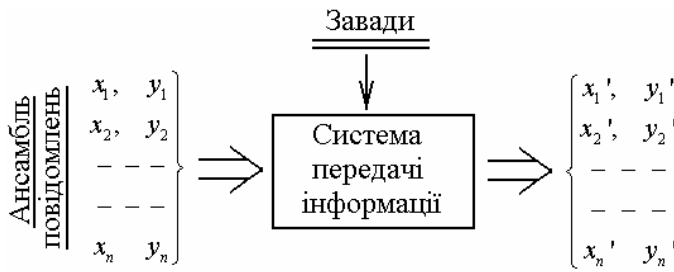


Рис. 1.6. Модель системи передавання інформації з використанням ансамблю повідомлень

Під ансамблем повідомлень розуміють множину можливих повідомлень з відомими їхніми ймовірнісними характеристиками:  $\{X, p(x)\}$ , де  $X = \{x_1, x_2, \dots, x_n\}$  – множина можливих повідомлень джерела;  $i = 1, 2, \dots, m$ , де  $m$  – об'єм алфавіту,  $p(x_i)$  – ймовірності появи повідомлень, причому  $p(x_i) \geq 0$ . Оскільки ймовірності повідомлень становлять собою повну групу подій, то їхня сумарна ймовірність рівна одиниці:

$$\sum_{i=1}^m p(x_i) = 1.$$

Кожне повідомлення несе в собі певну кількість інформації. Визначимо кількість інформації, що міститься у повідомленні  $x_i$ , вибраному з ансамблю повідомлень джерела  $\{X, p(x)\}$ . Одним з параметрів, що характеризує це повідомлення, є ймовірність його появи –  $p(x)$ , тому необхідно припустити, що кількість інформації  $I(x_i)$  у повідомленні  $x_i$  є функцією  $p(x_i)$ . Ймовірність появи двох незалежних повідомлень  $x_1$  і  $x_2$  рівна добутку ймовірностей  $p(x_1, x_2) = p(x_1) \cdot p(x_2)$ , а інформація, що міститься в них, має властивість адитивності:

$$I(x_1, x_2) = I(x_1) + I(x_2). \quad (1.1)$$

Тому для оцінення кількості інформації за Шенноном запропонована логарифмічна міра:

$$I(x_i) = \log \frac{1}{p(x_i)} = -\log p(x_i). \quad (1.2)$$

Водночас найбільша кількість інформації містить найменш ймовірні повідомлення, а кількість інформації у повідомленні про достовірну подію рівна нулю. Оскільки всі логарифми пропорційні, то вибір основи визначає одиницю інформації:  $\log_a x = \log_b x / \log_b a$ . Залежно від основи логарифму використовують такі одиниці інформації: 2 – [біт] (binary digit – двійкова одиниця), використовую-

ють в інформаційних процесах ЕВМ та інших пристроях, що функціонують на основі двійкової системи числення; 10 – [діт] (decimal digit – двійкова одиниця), використовують при аналізі процесів у пристроях, що функціонують на основі десяткової системи числення. Двійкове слово з восьми символів містить байт інформації, 1024 байтів утворюють кілобайт (Кбайт), 1024 Кбайтів – мегабайт (Мбайт) і 1024 Мбайтів – гігабайт (Гбайт), при цьому  $1024 = 2^{10}$ .

Біт є не лише одиницею кількості інформації, але й одиницею вимірювання ступеня невизначеності. Тут розуміють невизначеність, яка міститься в одному досліді, що має два рівноймовірні результати.

Як одиниця статистичної міри інформації *біт* – це кількість інформації, яка знімає невизначеність щодо реалізації однієї з двох рівноймовірних ( $p_0=0,5$ ,  $p_1=0,5$ ) незалежних подій.

Середню кількість інформації для всієї сукупності повідомлень можна отримати шляхом усереднення за всіма незалежними подіями:

$$I(x) = -\sum_{i=1}^m p_i \log p_i. \quad (1.3)$$

Кількість інформації у повідомленні, що складається з  $n$  нерівноймовірних його елементів рівна за Шенноном:

$$I(x) = -n \sum_{i=1}^m p_i \log p_i. \quad (1.4)$$

Для випадку незалежних рівноймовірних подій кількість інформації визначають за Хартлі (1928 р.) так:

$$I(x) = -n \sum_{i=1}^m p_i \log p_i = -n \cdot m \left( \frac{1}{m} \log \frac{1}{m} \right) = n \log m. \quad (1.5)$$

Середнє значення кількості інформації на одне повідомлення за Шенноном визначає ентропію. Сутність ентропії сформулював німецький фізик Больцман (1871р.). Використовуючи статистичну модель ансамблю молекул газу, він увів цей параметр як *міру хаосу* в ансамблі молекул. З цієї позиції поняття інформація можна класифікувати, як *ентропію з протилежним знаком*.

На основі описаної моделі ансамблю подій *ентропія* – це математичне очікування випадкової величини  $I(x)$  визначеної на ансамблі  $\{X, p(x)\}$ , тобто вона характеризує середнє значення кількості інформації, що приходить на один символ (повідомлення):

$$H(x) = M[I(x)] = \frac{I(x)}{n} = \sum_{i=1}^m p(x_i) \log p(x_i). \quad (1.6)$$



Наочним прикладом для цього випадку є задача про ентропію системи двох альтернативних подій з ймовірностями  $p_1$  і  $p_2$ . Ентропія у цьому випадку рівна:

$$\begin{aligned} H(x) &= -p_1 \cdot \log_2 p_1 - p_2 \cdot \log_2 p_2 = \\ &= -p_1 \cdot \log_2 p_1 - [1 - p_1] \log_2 [1 - p_1] = \begin{cases} 1; & p_1 = p_2 = 1/2 \\ 0; & p_1 = 0, p_2 = 1 \\ & \text{або } p_1 = 1, p_2 = 0 \end{cases} \end{aligned} \quad (1.7)$$

Ентропія повідомлень має такі властивості: 1) вона є величина дійсна, обмежена, не від'ємна, неперервна в інтервалі  $0 \leq p \leq 1$ ; 2) ентропія максимальна для рівноймовірних подій; 3) ентропія для детермінованих подій рівна нулю; 4) ентропія системи двох альтернативних подій змінюється від нуля до одиниці.

Ентропія чисельно збігається з середньою кількістю інформації, але принципово відмінна від неї, оскільки  $H(x)$  визначає середню невизначеність стану джерела й є його об'єктивною характеристикою. Вона може бути обчислена апіорно, тобто до отримання повідомлення за наявності статистики повідомлень. Одночасно,  $I(x)$  визначається апостеріорно, тобто після отримання повідомлення. З отриманням інформації про стан системи ентропія знижується.

### 1.3. Сигнали. Форми подання інформації [1]

Щоб передати інформацію від джерела до приймача, повідомлення перетворюють у сигнали. За означенням, сигнал – це *зміна фізичної величини, використовуваної для пересилання даних*. Як відомо, сигнал утворюється на основі деякого фізичного об'єкта (процесу) – електромагнітні чи акустичні коливання, електрична напруга, струм та ін., який традиційно називають “носієм”. Може відбуватися зміна одного або декількох параметрів (амплітуди, фази, частоти) за визначеним законом передавання інформації. Вважають, що сигнал – *це матеріально-енергетичне втілення повідомлення*. Він може перетворюватися без зміни змісту інформації з однієї фізичної величини в іншу, зручнішу для оброблення комп'ютером. Зміну параметрів фізичної величини за законом переданого повідомлення називають *модуляцією*.

Сигнали класифікують на: 1) випадкові і детерміновані (за ступенем визначеності очікуваних значень); 2) неперервні та дискретні (за структурою часової змінної); 3) адреси, дані, сигнали керування (за роллю передаваної інформації); 4) низькочастотні та високочастотні, вузькосмугові, широкосмугові (за спектральним представленням); 5) кодовані, декодовані, дискретизовані (за способом перетворення); 6) потенціальні та імпульсні (за характером зміни кодованих сигналів у синхронні моменти часу).

Спосіб формалізованого опису різноманітних сигналів називають *поданням інформації*. Зазвичай у теорії інформації розглядають не фізичне, а математичне подання сигналів, тобто їхній опис за допомогою формул, графіків, функцій, розподілу ймовірностей тощо.

Найбільш поширеними способами подання сигналів є *часовий, спектральний, статистичний*. Виділяють такі різновидності сигналів, які описують за часовою функцією  $y(t)$ :

- 1) неперервна функція неперервного аргументу в інтервалі часу  $0 \leq t \leq t_k$  (див.: рис. 1.7, а);
- 2) дискретна функція неперервного аргументу (див.: рис. 1.7, б). Значення, отримані функцією  $y(t)$  утворюють дискретний ряд чисел  $y_i$  ( $i = 1, 2 \dots k$ ). Значення аргументу варіює в інтервалі часу  $0 \leq t \leq t_k$ . Перетворення неперервної функції  $y(t)$  у дискретну множину значень  $y_i$  називають “квантуванням за рівнем”;
- 3) неперервна функція дискретного аргументу (див.: рис. 1.7, в). Значення функції  $y(t)$  визначають тільки на дискретній множині  $t_i$  ( $i = 1, 2 \dots k$ ). Функція  $y(t_i)$  може набувати довільних значень у заданому діапазоні. Перетворення функції  $y(t)$  неперервного аргументу  $t$  у функцію  $y(t_i)$  дискретного аргументу  $t_i$ , називають *дискретизацією (квантуванням) у часі*;
- 4) дискретна функція дискретного аргументу (див. рис. 1.7, г). Значення, що приймають функція й аргумент, утворюють дискретні ряди чисел  $y_0, y_1, \dots, y_k$  і  $t_0, t_1, \dots, t_k$ .

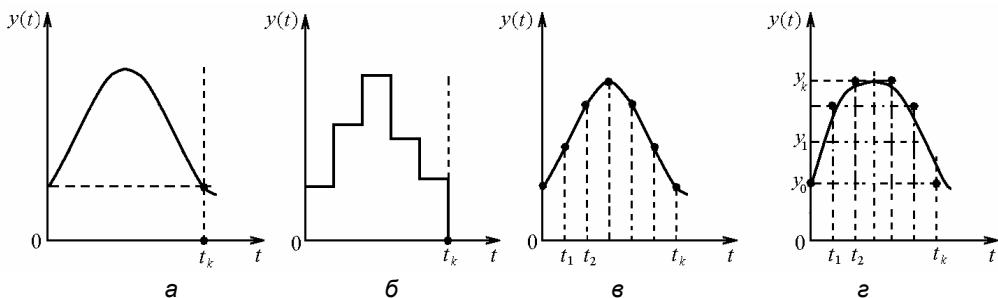


Рис. 1.7. Математичне представлення сигналів

Перший тип сигналів описує неперервні (аналогові) сигнали; другий і третій – дискретно-неперервні, а четвертий – чисто дискретні. Сумісне застосування дискретизації та квантування дає змогу перетворювати неперервну функцію в чисто дискретну.

Згідно з теоремою Котельникова, сигнал, описуваний функцією з обмеженим спектром, визначається своїми дискретними значеннями, які відраховують через інтервал часу  $t = 1/2 F_C$ , де  $F_C$  – ширина спектра. Отже, сигнал  $y(t)$  можна

передавати окремими миттєвими значеннями, які відраховують через кінцевий інтервал часу. За цими значеннями комп'ютер повністю відновлює первинний неперервний сигнал.

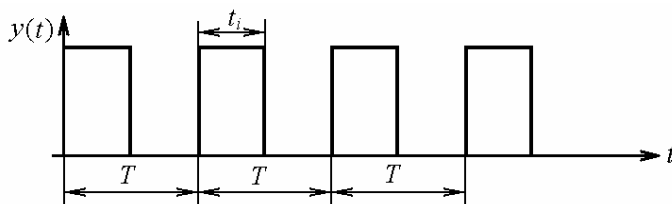


Рис. 1.8. Часоімпульсне представлення сигналу  $y(t)$  прямокутними імпульсами

До дискретно-неперервних функцій відносять також часоімпульсне представлення первинного сигналу  $y(t)$  прямокутними імпульсами з неперервним інформативним параметром,  $t_i/T$ , де  $t_i$  – тривалість імпульсів, пропорційна значенню сигналу;  $T$  – період імпульсів (див.: рис. 1.8).

#### 1.4. Аналогова та цифрова форми подання інформації

**Аналогова форма подання інформації.** Аналоговою формою подання (АФП) інформації людство користувалося достатньо давно, не детермінуючи цієї назви. Однак сам термін (АФП) сформувався внаслідок усвідомлення поняття інформації як предмета вивчення науки та у зв'язку з реалізацією математичних моделей в аналогових вчислювальних машинах. Відомо, що такі моделі використовують як методичну основу аналогових систем диференціальних рівнянь, які описують різноманітні природні явища та процеси [2, 3].

Загалом інформативним для технічних систем може бути один або декілька параметрів довільного фізичного процесу. Зокрема, для гармонічних сигналів інформативними можуть бути амплітудне значення, частота, фаза або їхні комбінації. Особливістю є те, що для технічних систем зазвичай всі значення АФП сигналу мають бути однаково інформативними. Ця обставина зумовлює суттєву особливість АФП – принципову *незахищеність* від впливу завад. Об'єми інформації, що її передають за допомогою АФП, через значимість кожного значення неперервного сигналу в технічних системах можуть бути дуже великими й обмежені лише заданою точністю реалізації функції технічної системи.

Переважно в аналоговій схемотехніці використовують два види інформаційних сигналів: періодичні (модульовані / не модульовані) та неперіодичні, причому незалежною змінною в обох випадках є час.

Під *модульованим* розумітимемо деякий сигнал, один з параметрів якого змінюється відповідно до передаваної інформації. Наприклад, для гармонічного сигналу  $A(t) = A_0 \sin(\omega t + \varphi_0)$  можливими є три види модуляції:

$$A(t) = A_0(t) \sin(\omega(t) + \varphi_0) \quad (\text{амплітудна}),$$

$$A(t) = A_0(t) \sin[\omega(t) + \varphi_0] \quad (\text{частотна}),$$

$$A(t) = A_0(t) \sin[\omega(t) + \varphi(t)] \quad (\text{фазова}).$$

Аналогічні види модуляції можливі і для негармонічних періодичних сигналів.

Неперіодичні сигнали можуть бути відображені сумою окремих гармонічних коливань. Апаратом таких перетворень є інтеграл Фур'є:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt.$$

При частотному відображенні неперіодичного сигналу зазвичай уводять поняття спектра сигналу – як сукупності (суми) гармонічних складових з неперервною змінною частотою й амплітудою, що задовольняє приведеному інтегралу Фур'є.

**Цифрова форма подання інформації** має принципові відмінності від аналогової саме в тому, що з метою забезпечення довільної наперед заздалегідь точності в області цифрових сигналів (за  $U$  чи  $I$ ) уведено спеціальні зони, значення сигналів яких або рівнозначні (незалежно від величини зони), або не можуть бути наявні у принципі (див.: рис. 1.9).



Рис. 1.9. Схема рівнів цифрового представлення інформації згідно з угодою

Це не означає, що струм або напруга фізично неможливі в забороненій зоні або фізично не відрізняються одне від одного у дозволений. Насправді є умовна

“домовленість” між розробниками пристроїв про те, як необхідно інтерпретувати певні значення сигналів у пристроях. Зазначимо, що введення дозволених і заборонених зон еквівалентне введенню надлишкової інформації у передаваний сигнал. Саме ця обставина дає можливість підвищити достовірність цифрового повідомлення, але й обмежує його інформаційний об'єм.

Кількість дозволених і заборонених зон може бути довільною, її визначають за прийнятою системою числення. Разом з тим кожній з дозволених зон умовно присвоюють значення деякої цифри. Однак на практиці домовленість про представлення інформації та реалізація цієї угоди в технічних пристроях зовсім не є одне одне і те ж саме. Для прикладу, фізичні процеси в електронних приладах протікають неперервно і транзистор “не знає”, що він має реалізувати деяке цифрове представлення, тому виникає обмеження на кількість зон при цифровій формі представлення інформації. За сучасного рівня розвитку електроніки компроміс між бажаним і можливим реалізований на двійковому поданні інформації (саме через схемотехнічні можливості, а не за бажанням розробників).

При двійковому поданні інформації наявні дві дозволені зони, яким можуть бути присвоєні значення логічного “0” і “1”, та одна заборонена зона, обмежена між дном зони високих рівнів та “стелею” зони низьких рівнів. Розміри цих зон визначають при конкретній технічній реалізації пристрою.

Ще є одна важлива обставина. Деякий цифровий пристрій має не лише формувати двійкові рівні, але й розпізнати їх при прийманні. А це означає, що в угоді про цифрову форму подання інформації необхідно ввести ще одну зону – зону порогової напруги – деякої напруги, величина якої не залежить від величини логічних рівнів. Конкретну величину порогової напруги визначають за схемою цифрового елемента та використовуваними в ньому характеристиками напівпровідникових приладів.

Перевищення порогової напруги (за “нульової” вхідної напруги) зазвичай веде до відкриття транзистора формувача рівня в логічному елементі (відкриваюча завада). При “одиничній” вхідній напрузі можлива закриваюча завада. Отже, з введенням порогової напруги заборонена зона розпадається на дві зони: зону відкриваючих і зону закриваючих завад. Величину цих зон визначають за заданою завадостійкістю – здатністю логічного елемента достовірно відрізняти високі та низькі рівні сигналів за наявності завад. Необхідно говорити про *статичну* та *динамічну завадостійкість*, розуміючи, що при визначенні величини статичної завадостійкості не враховують часові параметри завади (час дії завади суттєво більший від часу перемикання цифрового елемента). Якщо час дії завади порівняний з часом перемикання цифрового елемента, то говорять про *динамічну завадостійкість*.

Зазначимо, що розбиття на зони може бути реалізоване до різних конфігурацій сигналів, наявних у часі. Насамперед за амплітудою (амплітудна форма), або за часом (часоімпульсна форма). Друга форма забезпечує велику завадостійкість, однак веде до достатньо громіздких технічних рішень, не знайшовши сьогодні застосування.

У разі поданні інформації амплітудою деякого сигналу суттєве значення мають і його часові параметри. З одного боку вони визначають швидкодію елементів комп'ютера, з іншого—від цифрового сигналу. Тому серед цифрових сигналів виділяють імпульси та потенціали. *Імпульс* – цифровий сигнал, характеристикою якого є його тривалість. Вона є незмінною, однак може змінюватися при переході від елемента до елемента, в середньому залишаючись постійною. *Потенціал* – цифровий сигнал, тривалість якого може бути довільною.

Прикладом описаних видів цифрових сигналів є імпульси, які застосовують у комп'ютерній електроніці. На рисунку 1.10 зображене часове розгортання *тактових імпульсів*, які генеруються у кожному комп'ютері для синхронізації процесів оброблення інформації.

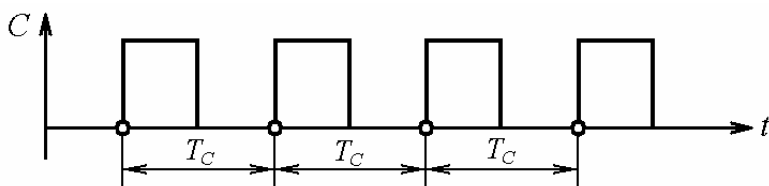


Рис. 1.10. Часова діаграма імпульсів, генерованих генератором тактових імпульсів комп'ютера

Початок кожного імпульсу називають *тактовим моментом*. Часовий інтервал між двома сусідніми імпульсами  $C$  називають *машинним тактом*  $T_C$ . На початку кожного імпульсу  $C$  змінюється інформація на входах елементів і вузлах комп'ютера.

Генератор, що виробляє такі імпульси називають *генератором синхронізуючих імпульсів* (ГСІ) (або *тактових—ГТІ*), а самі імпульси *синхронізуючими*, або *синхроімпульсами*. Амплітуда і полярність імпульсу  $C$  залежать від фізичних принципів будови машини. Принцип подання інформації на входи елементів і вузлів у тактові моменти називають *дискретизацією сигналів у часі*.

У комп'ютерній схемотехніці застосовують два основні види двійкових сигналів: потенціальні та імпульсні (див.: рис. 1.11).

Сигнал, який змінюється лише в тактові моменти часу назвали *потенціальним*. Сигнал, який наростає в тактовий момент, а спадає в межах цього такту, назвали *імпульсним*. Тривалість потенціального сигналу дорівнює (або кратна) тривалості машинного такту.

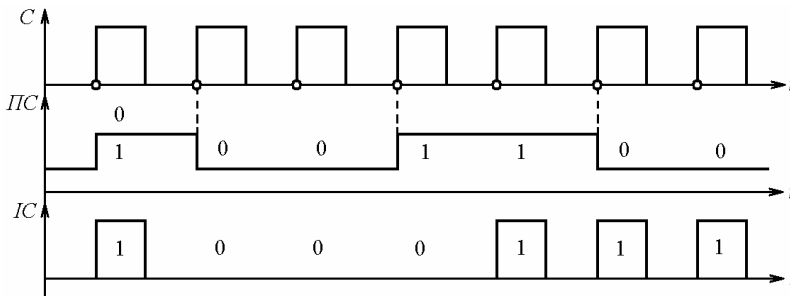


Рис. 1.11. Діаграми потенціальних та імпульсних сигналів

У логіці значення двійкового сигналу та відповідної змінної  $X$  кодують символами 0 ("0") та 1 ("1"). Напругу, що відображає "1" позначають  $U^1$ , а символ "0" –  $U^0$ . Розрізняють два способи кодування логічних сигналів  $X_i$  потенціальними сигналами: позитивний та негативний. За позитивного кодування (позитивна логіка (угода)); більший рівень напруги  $U_H$  з урахуванням знака відображає "1", а менший  $U_1$  – "0". Тобто  $X=1$ , якщо  $U^1 = U_H$  та  $X=0$  при  $U^0 = U_1$  (див.: рис. 1.12, а). При негативному кодуванні (негативна логіка) вищий рівень напруги  $U_H$  з урахуванням знака відображає "0", а нижчий  $U_1$  – "1". Тобто  $X=1$ , якщо  $U^1 = U_1$  та  $X=0$  при  $U^0 = U_H$  (див.рис.: 1.12, б).

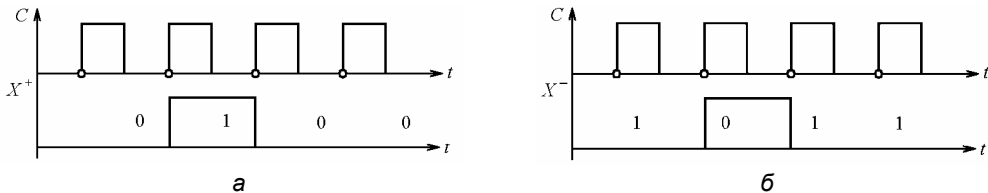


Рис. 1.12. Логічні угоди:

а – позитивна  $X^+$ ; б– негативна  $X^-$ 

Для імпульсних сигналів розрізняють два види кодування: наявність імпульсу відображає "1", а брак "0" (див.: рис. 1.13).

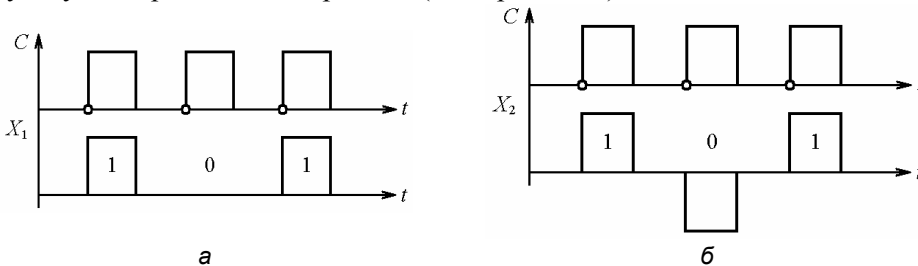


Рис. 1.13. Імпульсне кодування:

а – першого роду; б – другого роду

Сучасна цифрова схемотехніка головно користується потенціальними сигналами.

### 1.5. Принципи організації та структура комп'ютера

**Комп'ютер** (*computer*,англ.); *computator*,(лат.)– обчислювач; *computatrum* (лат.) – рахувати) – програмнокерований пристрій для оброблення інформації. Конструктивно це може бути механічний, електромеханічний або немеханічний (електронний) пристрій, призначений для проведення обчислень, які можуть відбуватися дискретно або безперервно у часі. У вузькому значенні – електронний цифровий програмований пристрій (*електронна обчислювальна машина*) для проведення обчислень заздалегідь визначеним алгоритмом. Сьогодні практично всі наявні комп'ютери є електронно-обчислювальними машинами. *Структура* комп'ютера – це *сукупність його функціональних елементів та зв'язків між ними*. Схематично її зображають як структурні схеми, за допомогою яких можна зробити опис комп'ютера на будь-якому рівні деталізації. Під *елементами* потрібно розуміти набір простих схем (типу узгодження, допоміжних), а також основних функціональних вузлів послідовного та комбінаційного типів, арифметико-логічний пристрій, аналого-цифрові перетворювачі, пристрої пам'яті та ін.

Розглядаючи питання про принципи роботи комп'ютера та його структуру не можна обійти поняття “архітектура” комп'ютера, оскільки саме вона визначає внутрішню сутність зв'язків у комп'ютері. Його вперше ввела компанія IBM в кінці 1950-х років, щоб описати рівень сумісності певного сімейства комп'ютерів, кожний з яких виконує одні і ті ж команди, але відрізняються за можливостями реалізації певної програми. З того часу це поняття зазнає різних дефініцій розробниками комп'ютерних систем, відображаючи таким чином їхню приналежність до відповідної галузі комп'ютерної індустрії. Наприклад, одне з них: “*архітектура – це логічна організація комп'ютера з позицій програміста*” – однобоко відображає взаємозв'язки в комп'ютері, що наявні лише на абстрактному рівні. Найбільш прийнятним і повним є означення *архітектури*, як *деякого абстрактного представлення, яке відображає його структурну, схемотехнічну та логічну організацію*. Архітектура визначає принципи дії, інформаційні зв'язки та взаємне з'єднання основних логічних вузлів комп'ютера, оперативного запам'ятовуючого пристрою (ОЗП), зовнішніх ЗП і периферійних пристроїв (ПП), а також набір і доступність регістрів, організацію і способи адресації пам'яті, способи представлення і формати даних, набір команд, оброблення переривань. Найбільш прийнятними на сьогодні є два типи архітектури, які поділяють за принципом розділення пам'яті. Перший тип – Гарвардська архітектура. Він передбачає розділення програм і пам'яті даних, а другий – фон-



Нейманівська (1945 р.) – має характерною ознакою сумісне збереження програм і даних. В основі побудови переважної кількості сучасних комп'ютерів є Нейманівська архітектура. На базі цих принципів було створено перші два покоління комп'ютерів, однак головні з них використовують і в сучасних поколіннях комп'ютерних систем. Основні принципи Нейманівської машини такі: використання двійкової системи числення; програмне управління; пам'ять машини використовують не лише для збереження даних, але й програм; принцип адресності: елементи пам'яті мають адреси, які послідовно пронумеровані; можливість умовного переходу в процесі виконання програм до довільної ділянки кодів.

Структурна схема типового персонального комп'ютера зображена на рисунку 1.14.

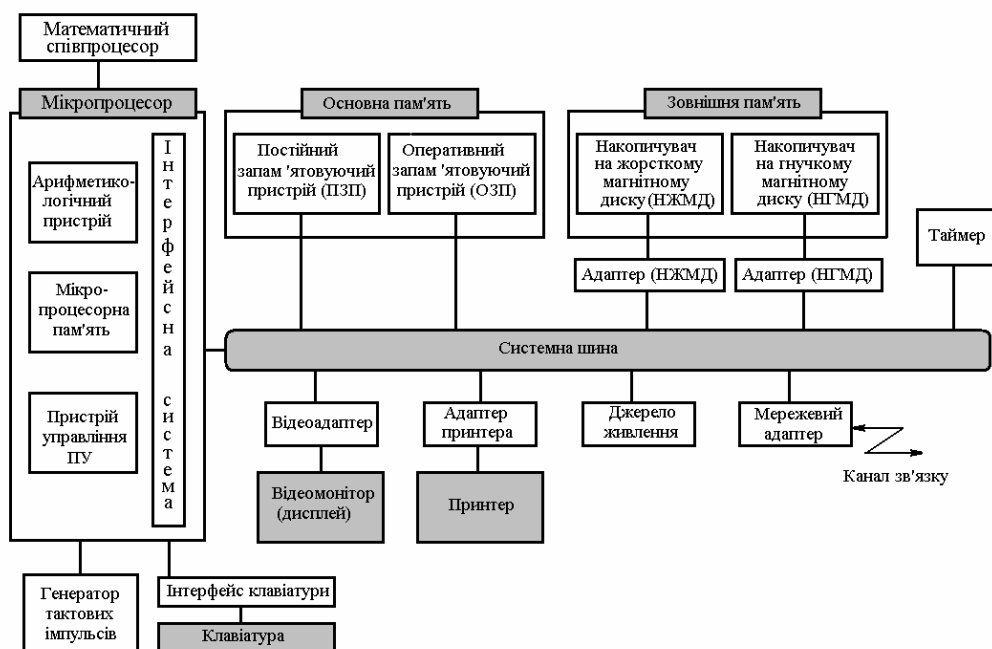


Рис. 1.14. Структурна схема персонального комп'ютера

**Мікропроцесор (МП).** Мікропроцесор – центральний пристрій ПК, призначений для управління роботою всіх блоків комп'ютера та виконання арифметичних і логічних операцій над інформацією. МП виконує такі функції: обчислення адрес команд і операндів; вибірку і дешифрацію команд з основної пам'яті (ОП); вибірку даних з ОП, регістрів МПП і регістрів адаптерів зовнішніх пристроїв (ЗП); приймання і оброблення запитів і команд від адаптерів на обслуговування ЗП; оброблення даних та їхній запис в ОП, регістри СПП і регістри

адаптерів ЗП; вироблення управляючих сигналів для всіх інших вузлів і блоків ПК; перехід до чергової команди.

**Системна шина.** Системна шина – основна інтерфейсна система комп'ютера, що забезпечує спраження та зв'язок усіх його пристроїв між собою. Системна шина включає: 1) кодову шину даних (КШД), що містить провідники і схеми для паралельного передавання всіх розрядів коду адреси комірки основної пам'яті або порту введення-виведення зовнішнього пристрою; 2) кодову шину адреси (КША), що містить провідники і схеми спраження для паралельного передавання всіх розрядів коду адреси комірки основної пам'яті або порту введення-виведення зовнішнього пристрою; 3) кодову шину інструкцій (КШІ), що містить провідники і схеми спраження для передавання інструкцій (управляючих сигналів, імпульсів) у всі блоки машини; 4) шину живлення, що містить провідники і схеми спраження для під'єднання блоків ПК до системи енергоживлення.

Системна шина забезпечує три напрямки передавання інформації: між МП і основною пам'яттю; між МП і портами введення-виведення зовнішніх пристроїв; між основною пам'яттю і портами введення-виведення зовнішніх пристроїв (у режимі прямого доступу до пам'яті).

Усі блоки, а точніше, їхні порти введення-виведення, через певні уніфіковані роз'єми під'єднуються до шини однаково: безпосередньо або через контролери (адаптери). Управління системною шиною здійснюється МП або безпосередньо, або, що найчастіше, через додаткову мікросхему контролера шини, що формує основні сигнали управління. Обмін інформацією між зовнішніми пристроями і системною шиною виконується з використанням ASC II-кодів.

**Основна пам'ять.** Основна пам'ять (ОП) призначена для збереження та оперативного обміну інформацією з іншими блоками машини. Вона містить два типи запам'ятовуючих пристроїв: постійний запам'ятовуючий пристрій (ПЗП) і оперативний запам'ятовуючий пристрій (ОЗП). ПЗП (ROM – **Read Only Memory**) призначений для збереження незмінної програмної і довідникової інформації; дає змогу оперативно лише зчитувати інформацію, що зберігається в ньому; ОЗП (RAM – **Random Access Memory**) призначений для оперативного запису і зчитування інформації (програм і даних), що безпосередньо бере участь в інформаційно-вчислювальному процесі, який виконується ПК у плинний період часу. Головною перевагою ОП є її висока швидкодія і можливість звертання до кожної комірки пам'яті окремо (прямий адресний доступ до комірки). Недоліком цього виду пам'яті є її енергозалежність, тобто зникнення оброблюваної інформації після вимкнення електроживлення.

Крім основної пам'яті на системній платі ПК є також енергонезалежна пам'ять CMOS RAM (**Complementary Metal-Oxide semiconductor RAM**), що

постійно живиться від свого акумулятора; в ній зберігається інформація про апаратну конфігурацію ПК, яка перевіряється за кожного ввімкнення системи.

**Зовнішня пам'ять.** Зовнішню пам'ять (ЗП) відносять до зовнішніх пристроїв комп'ютера; вона призначена для довготривалого збереження довільної інформації, яка коли-небудь стане необхідною для розв'язування задач. Зокрема, у ЗП зберігається все програмне забезпечення комп'ютера. ЗП має різні види запам'ятовуючих пристроїв, але найбільш поширеними є показані на структурній схемі накопичувачі на жорстких та гнучких магнітних дисках. Призначення цих накопичувачів – збереження великих об'ємів інформації, запис і видавання інформації в ОЗП за запитом. Розрізняють НЖМД і НГМД конструктивно, за об'ємами збережуваної інформації, часом її пошуку, запису і зчитування. У ролі пристроїв ЗП використовують також накопичувачі на оптичних дисках (CD ROM – Compact Disk Read Only Memory) та рідше стримери. В останні роки великої популярності набули пристрої флеш-пам'яті.

**Джерело живлення.** Джерело живлення – блок, що містить системи автономного і мережевого енергоживлення комп'ютера та забезпечує живленням з номінальними параметрами (напруга, сила струму, частота та ін.) всі функціональні блоки, вузли та системи комп'ютера.

**Таймер.** Таймер – внутрішній годинник реального часу, що забезпечує при потребі зчитування поточного моменту часу. Він живиться від автономного джерела живлення – акумулятора, а отже, при від'єднанні машини від мережі продовжує працювати.

**Генератор тактових імпульсів.** Результати функціонування цифрових схем, які входять до складу комп'ютера, залежать від порядку виконання певних дій, операцій. Для контролю часових співвідношень, щоб забезпечити синхронізацію у цифрові схеми влаштовують тактові генератори. Тактовий генератор – це схема, яка виробляє серію імпульсів. Всі імпульси однакові за тривалістю. Інтервали між послідовними імпульсами також однакові.

У комп'ютері за час одного такту може реалізуватися багато подій. Якщо вони мають відбуватися у певному порядку, то такт потрібно розділити на підтакти. Щоб досягнути кращого розділення, ніж в основного тактового генератора, необхідно зробити розгалуження від задаючої лінії тактового генератора і встановити схему з певним часом затримання. Таким чином породжується вторинний сигнал тактового генератора, який зсунутий по фазі щодо первинного (див.: рис.1.15,а). Часова діаграма (див.:рис. 1.15, б) забезпечує чотири початки відліку часу для дискретних подій: 1) наростаючий фронт C1; 2) задній фронт C2; 3) наростаючий фронт C2; 4) задній фронт C2.

Тактові генератори можуть бути синхронними. У цьому разі час стану з високим рівнем імпульсу рівний часу стану з низьким рівнем імпульсу (див.: рис. 1.15, б). Щоб отримати асинхронну серію імпульсів потрібно зсунути сиг-

нал задаючого генератора, використовуючи лінію затримування. Потім потрібно з'єднати отриманий сигнал з початковим сигналом за допомогою логічної функції І (див.: рис. 1.15, в, сигнал С).

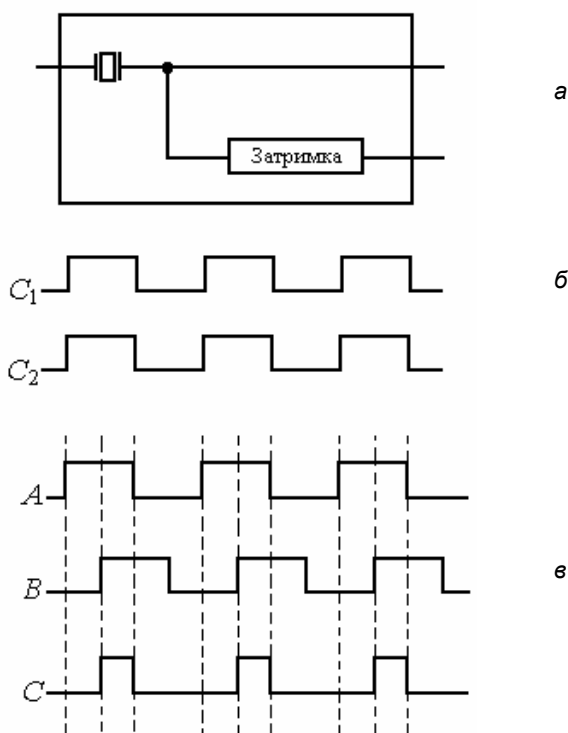


Рис. 1.15. Тактовий генератор (а); часова діаграма для тактового генератора (б); породження асинхронних тактових імпульсів

**Зовнішні пристрої.** Зовнішні пристрої (ЗП) – найважливіша складова довільного висувального комплексу. Достатньо сказати, що за вартістю ЗП становить до 80–85% вартості всього ПК. ЗП ПК забезпечують взаємодію машини з оточуючим середовищем: користувачами, об'єктами управління та іншими комп'ютерами. До ЗП відносять: зовнішні запам'ятовуючі пристрої (ЗЗП), або зовнішню пам'ять ПК; діалогові засоби користувача; пристрої введення інформації; пристрої виведення інформації; засоби зв'язку і телекомунікацій. Діалогові засоби користувача включають: відеомонітор; пристрої мовного введення-виведення – засоби мультимедіа. До пристроїв введення інформації відносять: клавіатуру; графічні планшети; сканери; пристрої цілевказівки (графічні маніпулятори); сенсорні екрани. До пристроїв виведення інформації відносяться: принтери; графобудувачі (плотери).

Пристрої зв'язку і телекомунікації використовують для зв'язку з приладами та іншими засобами телекомунікації та для під'єднання ПК до каналів зв'язку, інших комп'ютерних мереж.

Мультимедіа – це комплекс апаратних і програмних засобів, які дають можливість людині спілкуватися з комп'ютером, використовуючи різні, природні для себе середовища: звук, відео, графіку, тексти, анімацію та ін.

**Додаткові ІМС.** До системної шини і до МП ПК разом з типовими зовнішніми пристроями можуть бути під'єднані і деякі додаткові ІМС, які розширюють та покращують функціональні можливості МП: математичний співпроцесор; контролер прямого доступу до пам'яті; співпроцесор введення-виведення; контролер переривань.

Математичний співпроцесор використовують для прискорення виконання операцій над двійковими числами з фіксованою та плаваючою комою, над двійково-кодованими десятковими числами для обчислення деяких трансцендентних, у т.ч. тригонометричних функцій. Математичний співпроцесор має свою систему команд і працює паралельно в часі з основним МП, під його безпосереднім управлінням. Прискорюються операції в десятки разів. Сучасні моделі МП, починаючи з 80486 DX, включають співпроцесор у свою структуру.

Контролер прямого доступу до пам'яті (DMA–Direct Memory Access) забезпечує обмін даними між зовнішніми пристроями й оперативною пам'яттю без участі МП, що суттєво підвищує ефективну швидкість ПК. Режим DMA дає змогу звільнити процесор від рутинного пересилання даних між зовнішніми пристроями і ОП, віддавши цю роботу контролеру DMA. МП у цей час може обробляти інші дані або іншу задачу у багатозадачній системі.

Співпроцесор введення-виведення за рахунок паралельної роботи з МП суттєво прискорює виконання процедур введення-виведення при обслуговуванні декількох зовнішніх пристроїв; звільняє МП від оброблення процедур введення-виведення, в тому числі реалізує і режим прямого доступу до пам'яті.

Контролер переривань обслуговує процедури переривання. Переривання – часове призупинення виконання одної програми з метою оперативного виконання іншої, у якийсь момент більш важливої (пріоритетної) програми. Контролер приймає запит на переривання від зовнішніх пристроїв, визначає рівень пріоритету цього запиту та видає сигнал переривання в МП. Мікропроцесор, отримавши цей сигнал призупиняє виконання поточної програми і переходить до виконання спеціальної програми обслуговування переривання, яке запросив зовнішній пристрій. Після завершення програми обслуговування відновлюється виконання перерваної програми. Контролер переривань програмований. Переривання виникають при роботі комп'ютера постійно. Достатньо сказати, що всі процедури введення-виведення інформації виконуються за перериваннями.

**Елементи конструкції ПК.** Конструктивно ПК виконані у вигляді центрального системного блока (див.: рис.1.16), до якого через роз'єми під'єднують зовнішні пристрої: додаткові блоки пам'яті, клавіатуру, дисплей, принтер та ін.

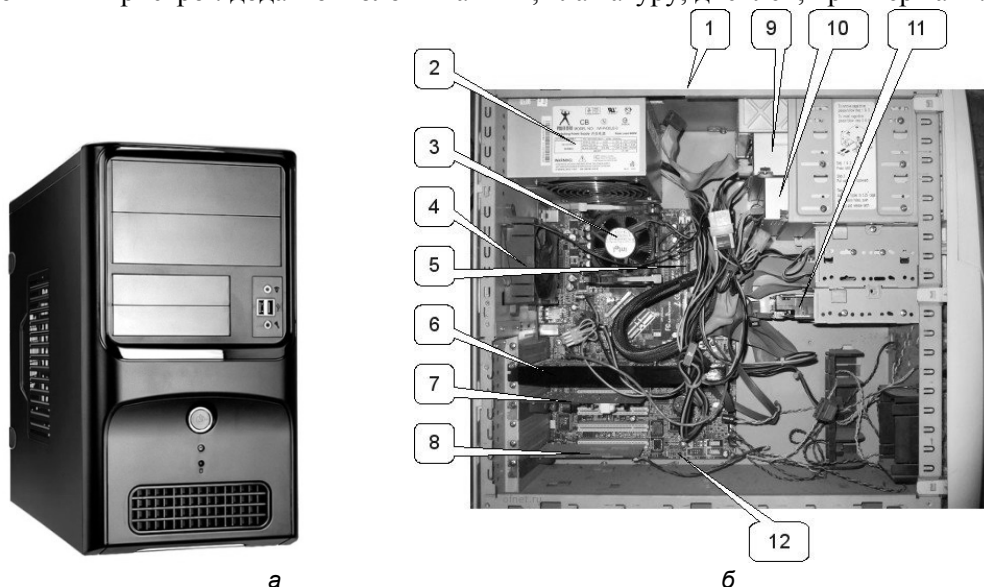


Рис. 1.16. Загальний вигляд типового системного блока: зовнішній вигляд (а) та зі знятою боковою кришкою (б)

**Системний блок** – металічний каркас (корпус) (1), який є основою для монтажу внутрішніх елементів та вузлів комп'ютера та містить системну плату (12); блок живлення (2); адаптери; один (іноді більше) накопичувач на жорсткому магнітному диску (НЖМД) (11); динамік; дисковод для компакт-дисків або інші накопичувачі (9, 10); модулі оперативної пам'яті (5); органи керування, а також захищає їх від зовнішнього впливу та механічних пошкоджень, підтримує необхідний температурний режим у середині системного блоку за допомогою корпусного вентилятора (4) всередині корпусу, що зазвичай видаляє тепле повітря з корпусу комп'ютера, зумовлюючи притік холодного повітря ззовні, екранує створені внутрішніми компонентами електромагнітні випромінювання та є основою для дальшого розширення системи. Відеокарта (відеоадаптер, відеоплата) (6) обробляє та виводить графічну інформацію на монітор. Вона містить свій спеціалізований графічний процесор, який обробляє 2D/3D графічну інформацію, що дає змогу знизити навантаження на центральний процесор. Модем (7) – пристрій зв'язку для перетворення сигналу за допомогою процесів модуляції та протилежного йому процесу демодуляції, що дає змогу комп'ютеру передавати дані по телефонній лінії; він є пристроєм узгодження у телекомунікаційних системах, системах автоматичного керування тощо. Моде-

ми, застосовувані в комп'ютерній техніці, бувають *внутрішні* (встановлені всередині системного блока) та *зовнішні* (встановлені ззовні). Через мережеву карту (8) комп'ютер під'єднується до локальної або глобальної мережі (зазвичай інтегрована на материнській платі).

Серед органів керування, переважно встановлених на передній панелі, можуть бути: вимикач електроживлення; кнопка загального скидання RESET; кнопка “сну”, яка дає змогу знизити енергоспоживання, коли комп'ютер не використовується; індикатори живлення та режимів роботи.

Із тильного боку системного блока розміщені штепсельні роз'ємні з'єднання – *порти* для під'єднання шнурів живлення і кабелів зв'язку із зовнішніми (встановленими поза системним блоком) пристроями. В середині системного блока розміщені плати сполучення пристроїв із центральним процесором та іншими пристроями на материнській платі (адаптери, контролери і плати розширення).

Усі комплектуючі системного блока комп'ютера умовно поділяють на дві категорії. Перша з них включає “життєво” необхідні: корпус, жорсткий диск, процесор, блок живлення, материнську плату, систему охолодження, оперативну пам'ять. Друга категорія – ті компоненти, без яких ПК може працювати, але його функціональність буде серйозно зменшена: відеокарта, привід оптичних дисків (CD, DVD, BluRay), TV-карта, аудіокарта, супутникова карта.

Найголовнішою частиною системного блока є *материнська плата* (*системна плата*), відома також як *головна плата* (*mainboard*) – *плата, яка містить основні компоненти комп'ютера*. Основу материнської плати становить багатшарова текстолітова (полімерна) конструкція з провідними металічними доріжками, як на поверхні, так і між шарами, засобом яких електричний струм передається між елементами плати. Це т. зв. *шиноорієнтована* плата. Отже, материнська плата є не лише несучою для головних компонентів, але й виконує комутативну функцію між ними за допомогою провідних шин. Вона реалізує схему мінімальної конфігурації, решта функцій реалізується за допомогою численних додаткових пристроїв. Усі компоненти з'єднуються шиною. У системній платі немає відеоадаптера, деяких видів пам'яті і засобів зв'язку з додатковими пристроями. Ці пристрої (плати розширення) додають до системної плати шляхом приєднання до шини розширення, яка є частиною системної плати.

Залежно від геометричних розмірів та практичного призначення комп'ютера системні плати класифікують за форм-фактором. Зокрема, горизонтальні: Desktop (533×419×152 мм), FootPrint (406×406×152 мм), SlimeLine (406×406×101 мм), UltraSlimeLine (381×352×75 мм). Вертикальні: MiniTower (152×432×432 мм), MidiTower (172×432×490 мм), BigTower (190×482×820 мм), SuperFullTower (різні розміри).

Перша системна плата була розроблена фірмою IBM і продемонстрована в серпні 1981 року (PC-1). У 1983 році з'явився комп'ютер зі збільшеною системною платою (PC-2). Максимум, що могла підтримувати PC-1 без використання плат розширення – 64К пам'яті. PC-2 мала вже 256К, але найважливіша відмінність полягала в програмуванні двох плат. Системна плата PC-1 не могла без корегування підтримувати найбільш могутні пристрої розширення, такі, як твердий диск і покращені відеоадаптери.

Схематична конструкція типової материнської плати зображена на рисунку 1.17. Взаємодія елементів забезпечується чипсетом, який складається, зазвичай з двох частин — північного моста (Northbridge) і південного моста (Southbridge). Північний і південний мости розміщені на окремих мікро-схемах.

Передусім саме північний і південний мости визначають особливості системної плати і те, які пристрої можуть під'єднуватися до неї. Іншими ключовими елементами є роз'єми для підключення центрального процесора, графічного адаптера, звукової плати, жорстких дисків, оперативної пам'яті. Крім них, на платі містяться резистори, конденсатори, що підтримують роботу кожної деталі. Живлення материнської плати, а отже, й всіх під'єднаних до неї пристроїв забезпечується блоком живлення, поєднаною з платою кабелями.

Всі основні електронні схеми плати і необхідні додаткові пристрої інтегровані в системну плату, або вони під'єднані до неї за допомогою слотів розширення. Загальний вигляд типової материнської плати поданий на рисунку 1.18.

На системній платі розміщені:

- **Сокет або гніздо центрального процесора.** За допомогою контактних ніжок чи пружних контактів процесор поєднується з сокетом. Здебільшого передбачається можливість заміни процесорів, проте зрідка центральний процесор припаюють до плати (BGA) (1).

- **Мікросхема BIOS** призначена для забезпечення первинної роботи комп'ютера. Вона містить інформацію про під'єднані пристрої, режими їхньої роботи, та надає користувачеві через графічний інтерфейс на моніторі змогу

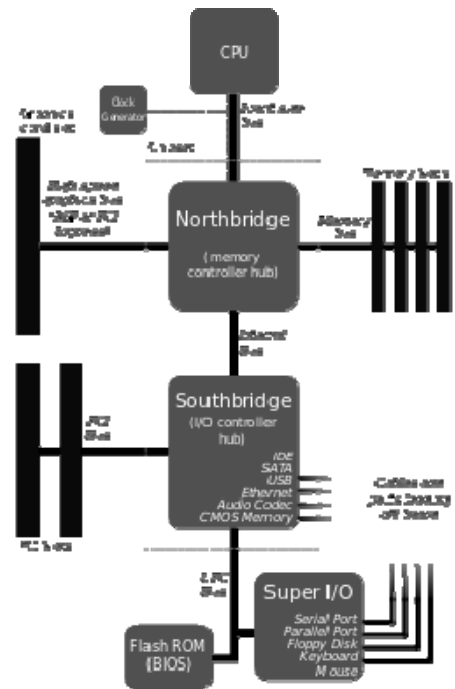


Рис. 1.17. Схематичне розміщення чипсетів на материнській платі



змінювати конфігурацію обладнання. *Батарея*, що міститься поряд, забезпечує живлення годинника, що служить для синхронізації пристроїв і забезпечує зручність користування комп'ютером. Наприклад, він визначає системний час і термін, упродовж якого користувач має змогу увійти в керування BIOS.

- **Слоти модулів оперативної пам'яті.** Модулі форм-фактора DIMM типу SDRAM, такі як DDR, DDR2 і новіші (слоти різні для кожного типу пам'яті). Найчастіше їх три-чотири, хоча на компактних платах їх зазвичай є лише один або два (2).



Рис. 1.18. Загальний вигляд типової материнської плати

- **Роз'єм відеокарти.** Спеціалізований роз'єм типу AGP або PCI-Express служить передусім для установлення відеокарти. Зазвичай він один, втім, останнім часом зустрічаються плати з двома, а то і з трьома відеороз'ємами. Також трапляються і системні плати без відеороз'ємів взагалі – їхні чипсети мають вбудоване графічне ядро і зовнішня графічна карта для них необов'язкова. В останньому випадку відеокарта використовує частину оперативної пам'яті, а не окрему відеопам'ять (3).

- **Слоти під'єднання додаткових карт розширення** стандартів PCI або PCI-Expressx1(раніше використовувалися слоти ISA). Через них під'єднують SSD-накопичувачі, контролери USB, WiFi-карти (4).

- **Інтерфейси Serial ATA** (раніше IDE) для під'єднання дискових накопичувачів –жорстких дисків і оптичних приводів. Також там може бути роз'єм для *floppy*-дисководу (3,5"-дискети). Всі дискові накопичувачі

під'єднуються до системної плати за допомогою спеціальних кабелів, які в розмовній мові називають "шлейфами" (5).

- **Роз'єми живлення** (основні два типи – 24-контактний ATX і 4-контактний ATX12V для додаткової лінії +12 V) і дво-, три- або чотирифазний модуль регулювання напруги VRM (Voltage Regulation Module), що складається з силових транзисторів, дроселів і конденсаторів. Цей модуль перетворює, стабілізує і фільтрує напругу, що подається від блока живлення (6).

- **Задня панель** з роз'ємами для під'єднання додаткових зовнішніх пристроїв — монітора, клавіатури і миші, мережевих-, аудіо- і USB-пристроїв тощо. Часто комплектувана заглушкою, що закриває невикористовуваний простір між роз'ємами і корпусом.

- Окрім описаних слотів і роз'ємів, на довільній системній платі є велика кількість допоміжних *джамперів (перемичок)* і *роз'ємів*: це можуть бути і контакти для під'єднання *системного динаміка, кнопок і індикаторів на передній панелі корпусу*, і *роз'єми для під'єднання вентиляторів*, і *контактні колодки для під'єднання додаткових аудіороз'ємів і роз'ємів USB і FireWire*.

Створюючи конструкторські розробки важливо є класифікувати системні плати за форм-фактором. *Форм-фактор системної плати*—стандарт, що визначає розміри системної плати для персонального комп'ютера, місця її кріплення до корпусу; розміщення на ній інтерфейсів шин, портів введення /виведення, сокета центрального процесора і слотів для оперативної пам'яті, а також тип роз'єму для під'єднання блоку живлення. Форм-фактор (як і будь-які інші стандарти) має рекомендаційний характер, проте більшість виробників намагається їх дотримуватися, оскільки ціною відповідності наявним стандартам є сумісність системної плати і стандартизованого устаткування (периферії, карт розширення) інших виробників. Застарілими вважають: Baby-AT; Mini-ATX; повнорозмірну плату AT; LPX. Сучасними вважають: ATX; microATX; Flex-ATX; NLX; WTX. Упроваджуваними вважаються: Mini-ITX і Nano-ITX; Pico-ITX; BTX, MicroBTX і PicoBTX. Також є системні плати невідповідні жодним з відомих форм-факторів. Зазвичай це зумовлено або тим, що комп'ютер є вузько спеціалізований, або бажанням виробника системної плати є самостійно виробляти і периферійні пристрої до неї. Причиною може бути також неможливістю використання стандартних компонентів ( т.зв. "бренд", наприклад Apple Computer, Commodore, Silicon Graphics, Hewlett Packard, Compaq частіше за інших ігнорували стандарти; крім того, в нинішньому вигляді розподілений ринок виробництва сформувався тільки до 1987 р., коли багато виробників вже створили власні платформи).

**Функціональні характеристики комп'ютера.** Головні функціональні характеристики комп'ютера : 1) продуктивність, швидкодія, тактова частота; 2) розрядність мікропроцесора та кодових шин інтерфейсу; 3) типи системного,

локальних і зовнішніх інтерфейсів; 4) тип і ємність оперативної пам'яті; 5) наявність, види і ємність кеш-пам'яті; 6) тип і ємність накопичувачів на жорстких магнітних дисках; 7) вид і ємність накопичувачів CD і DVD; 8) тип відеомонітора та відеоадаптера; 9) наявність і тип принтера; 10) наявність і тип модему; 11) наявність і види мультимедійних аудіо- та відеозасобів; 12) наявне програмне забезпечення та вид операційної системи; 13) апаратна і програмна сумісність з другими типами комп'ютерів; 14) можливість роботи у вичислювальній мережі та в багатозадачному режимі; 15) вартість, габарити і вага.

Конкретизуємо деякі з розглядуваних функціональних характеристик.

**Продуктивність, швидкодія, тактова частота.** Продуктивність сучасних комп'ютерів оцінюють зазвичай кількістю (у мільйонах) операцій за секунду. Головно одиницями вимірювання служать: МІПС (MIPS – Millions Instruction Per Second) – для операцій з числами, представленими у формі з фіксованою комою (крапкою); Мфлпс (MFLOPS – Millions of Floating point Operation per Second) – для операцій з плаваючою комою (крапкою).

Оцінення продуктивності комп'ютера завжди приблизне, оскільки зорієнтоване на деякі усереднені або, навпаки, на конкретні види операцій. У 70-ті роки двадцятого століття були розроблені усереднені набори операцій (суміші Гібсона) для різних типів задач: технічних, математичних, економічних та ін. в які різні команди входили у певному відсотковому співвідношенні. За сумішами Гібсона можна визначати середню швидкодію комп'ютерів для цих типів задач. Відомі й більш нові тести – тестові набори фірм-виробників для визначення швидкодії своїх виробів: показник iCOMP–Intel Corporative Microprocessor Performance (1992 рік) для МП фірми Intel; спеціалізовані тести для конкретних областей застосування комп'ютерів – Winstone97-Busines для офісної групи задач.

Для універсальних комп'ютерів, які виконують різноманітні завдання, ці оцінки будуть досить неточними, тому для характеристики ПК замість продуктивності зазвичай вказують тактову частоту, яка більш об'єктивно визначає швидкодію машини, оскільки кожна операція потребує для свого виконання певну кількість тактів. Знаючи тактову частоту, можна достатньо точно визначити час виконання довільної машинної операції.

Наприклад, у разі браку конвеєрного виконання команд та збільшення внутрішньої частоти в мікропроцесора тактовий генератор з частотою 100 МГц забезпечує виконання 20 млн коротких машинних операцій за секунду.

**Розрядність МП і кодових шин інтерфейсу.** Розрядність – це максимальна кількість розрядів двійкового числа, над яким одночасно може виконуватися машинна операція; чим більша розрядність, тим при інших рівних умовах буде вища продуктивність ПК. Розрядність МП визначається іноді за розрядністю його регістрів і кодової шини даних, а інколи за розрядністю кодових шин адреси.

**Типи системного, локальних і зовнішніх інтерфейсів.** Різні типи інтерфейсів забезпечують певні швидкості передавання інформації між вузлами машини, дають змогу під'єднати різну кількість зовнішніх пристроїв та їхні види, використовувати безпроводні канали зв'язку.

**Тип і ємність ОП.** Сучасні прикладні програми можуть працювати лише при певній ємності ОП. Збільшення ємності основної пам'яті у два рази, крім іншого, збільшує ефективну продуктивність комп'ютера при розв'язуванні складних задач приблизно в 1,41 рази (закон кореня квадратного). Різні типи ОП мають різні функціональні можливості.

**Наявність, види і ємність кеш-пам'яті.** Кеш-пам'ять – це буферна недоступна для користувача швидкодіюча пам'ять, що автоматично використовується комп'ютером для прискорення операцій з інформацією, що зберігається в більш повільних запам'ятовуючих пристроях. Наприклад, для прискорення операцій з основною пам'яттю зорганізується регістрова кеш-пам'ять всередині МП (кеш-пам'ять першого рівня), або зовні процесора на материнській платі (кеш-пам'ять другого рівня). Для прискорення операцій з дисковою пам'яттю організується кеш-пам'ять на комірках електронної пам'яті.

Наявність кеш-пам'яті збільшує продуктивність ПК приблизно на 20%.

**Апаратна і програмна сумісність з другими типами комп'ютерів.** Така властивість означає можливість використання програмного забезпечення і технічних продуктів не лише на конкретному комп'ютері, але й на інших типах машин.

**Можливість роботи у вичислювальній мережі та в багатозадачному режимі.** Багатозадачний режим дає змогу виконувати обчислення одночасно за декількома програмами (багатопрограмний режим) або для декількох користувачів (режим багатьох користувачів). Суміщення в часі роботи декількох пристроїв машини, що є можливим в такому режимі, дає можливість суттєво збільшити ефективну швидкодію комп'ютера.

## 1.6. Подання інформації у комп'ютерних системах

### 1.6.1. Арифметичні основи

**Системи числення.** Інформацію в комп'ютері прийнято кодувати у двійковій або двійково-десятковій системах числення.

**Система числення** – спосіб іменування та зображення чисел за допомогою символів, що мають певні кількісні значення. Залежно від способу зображення чисел системи числення поділяють на *позиційні* та *непозиційні*.

У *позиційній* системі кількісне значення кожної цифри залежить від її місця (позиції) у числі. У *непозиційній* системі цифри не змінюють свого кількісного значення при зміні їхнього розміщення у числі.

Кількість ( $P$ ) різних цифр, що їх використовують для зображення числа у позиційній системі числення, називають *основою системи числення*. Значення цифр лежить у межах від 0 до  $P-1$ . Узагальнено запис довільного змішаного числа в системі числення з основою  $P$  може бути репрезентований у вигляді полінома:

$$N = a_{n-1}P^{n-1} + a_{n-2}P^{n-2} + \dots + a_kP^k + \dots + a_0P^0 + a_{-1}P^{-1} + a_{-2}P^{-2} + \dots + a_{-s}P^{-s}. \quad (1.8)$$

Нижні індекси визначають розміщення цифри у числі (розряд): додатні значення індексів – для цілої частини числа ( $m$ -розрядів); від'ємні значення – для дробової ( $s$  розрядів).

Максимальне ціле число, яке може бути подане у  $m$  розрядах:

$$N_{\max} = P^m - 1. \quad (1.9)$$

Мінімальне значущє, не рівне 0 число, яке можна записати в  $s$  розрядах дробової частини:

$$N_{\min} = P^{-s}. \quad (1.10)$$

Маючи у цілій частині числа  $m$ , а у дробовій –  $s$  розрядів можна записати всього  $P^{m+s}$  різних чисел.

Двійкова система числення має основу  $P=2$  та використовує для подання інформації всього дві цифри – 0 і 1.

Щоб перевести числа з однієї системи числення в іншу застосовують правила, основані також і на співвідношенні (1.8). Наприклад, двійкове число 101110,101 рівне десятковому числу 46,625:

$$101110,101_2 = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = 46,625_{10}.$$

Практично переведення з двійкової системи у десяткову можна легко виконати, надписавши над кожним розрядом відповідну його вагу та склавши потім добутки значень отриманих цифр на їхні ваги. Двійкове число 01000001<sub>2</sub> рівне 65<sub>10</sub>. Вага 128 – цифра 0; 64 – 1; 32 – 0; 16 – 0; 8 – 0; 4 – 0; 2 – 0 і 1 – 1.

Отже, для переведення числа з непозиційної системи числення з довільною основою у десяткову систему числення можна використати співвідношення (1.8). Зворотнє переведення з десяткової системи числення у систему з другою основою безпосередньо за співвідношенням (1.8) достатньо ускладнене. Такий процес відбувається значно простіше, якщо попередньо перетворити окремо цілу частину  $N_u$  і дробову  $N_{op}$  частини виразу (1.8) до вигляду:

$$N_u = (((\dots (a_{n-1} \cdot P + a_{n-2}) \cdot P + \dots + a_2) \cdot P + a_1) \cdot P + a_0);$$

$$N_{op} = P^{-1} \cdot (a_{-1} + P^{-1} (a_{-2} + P^{-1} (a_{-3} + \dots + P^{-1} \cdot (a_{-s+1} + P^{-1} \cdot a_{-s}) \dots))).$$

Алгоритм переведення числа з десяткової системи числення у систему числення з основою  $P$ , оснований на цих виразах, дає змогу оперувати з числами у тій системі числення, з якої число переводиться, і може бути сформульований так: 1) при переведенні змішаного числа необхідно переводити його цілу і дробову частини окремо; 2) для переведення цілої частини числа її, а потім дробові частини отриманих часток від ділення, потрібно послідовно ділити на основу  $P$  до тих пір, доки чергова ціла частина частки не виявиться рівною нулю. Залишки від ділення, записані послідовно справа наліво, утворюють цілу частину числа у системі числення з основою  $P$ ; 3) для переведення дробової частини числа її, а потім дробові частини отримуваних добутків треба послідовно множити на основу  $P$  до тих пір, доки чергова дробова частина добутку не виявиться рівною нулю або не буде досягнута потрібна точність дробу. Цілі частини добутків, записані після коми послідовно зліва направо, утворюють дробову частину числа в системі числення з основою  $P$ .

Для прикладу розглянемо переведення з десяткової у двійкову систему числення числа 46,625. Переводимо цілу частину числа:  $46 / 2$  (залишок 0);  $23 / 2 = 11$  (залишок 1);  $11 / 2 = 5$  (залишок 1);  $5 / 2$  (залишок 1);  $2 / 2 = 1$  (залишок 0);  $1 / 2 = 0$  (залишок 1). Записуємо залишки послідовно справа наліво – 101110, тобто  $46_{10} = 101110_2$ . Переводимо дробову частину числа:  $0,125 \cdot 2 = 0,250$ ;  $0,250 \cdot 2 = 0,500$ ;  $0,500 \cdot 2 = 1,000$ . Записуємо цілі частини отриманих добутків після коми послідовно зліва направо – 0,101, тобто  $0,625_{10} = 0,101_2$ . Отже, кінцево:  $46,625_{10} = 101110,101_2$ .

**Подання чисел з фіксованою та плаваючою комами.** У вичислювальних машинах застосовують дві форми подання двійкових чисел: природна або форма з фіксованою комою (крапкою); нормальна форма або форма з плаваючою комою (крапкою). У формі подання з фіксованою комою всі числа зображають у вигляді послідовності цифр з постійним для всіх чисел положенням коми, яка відділяє цілу частину від дробової. Наприклад: у десятковій системі числення п'ять розрядів у цілій частині числа (до коми) і п'ять розрядів у дробовій частині числа (після коми); числа, записані в таку розрядну сітку мають вигляд: + 00721, 35500; +00000,000328; - 10301, 20260. Ця форма найбільш природна, проста, але має невеликий діапазон подання чисел, і тому зазвичай неприйнятна для обчислень. Діапазон значущих чисел  $N$  у системі числення з основою  $P$  при наявності  $m$  розрядів у цілій і  $s$  розрядів у дробовій частині числа (без урахування знаку числа) є таким:

$$P^{-s} \leq N \leq P^{m-s}.$$

Наприклад, при  $P=2$ ,  $m=10$  і  $s=6$  числа змінюються у діапазоні  $0,015 < N < 1024$ .

Якщо в результаті операції отримують число, що виходить за допустимі межі, то переповниться розрядна сітка і даліші обчислення втрачають зміст. У сучасних комп'ютерах природну форму подання використовують як допоміжну і лише для цілих чисел.

У формі подання з плаваючою комою кожне число зображають у вигляді двох груп цифр. Першу групу цифр названо *мантисою*, а другу – *порядком*, причому абсолютна величина мантиси має бути менше 1, а порядок – цілим числом. У загальному вигляді число у формі з плаваючою комою може бути подане так:

$$N = \pm MP^{\pm r},$$

де  $M$  – мантиса числа ( $M < 1$ );  $r$  – порядок числа (ціле число);  $P$  – основа системи числення.

Наприклад, розглянуті попередньо числа в нормальній формі можливо записати так:  $+0,721355 \cdot 10^3$ ;  $+0,328 \cdot 10^{-3}$ ;  $-0,103012026 \cdot 10^5$ .

Нормальна форма подання має величезний діапазон зображення чисел і є основною в сучасних комп'ютерах. Так, діапазон значущих чисел у системі числення з основою  $P$  при наявності  $m$  розрядів в мантиси і  $s$  розрядів у порядку буде:

$$p^{-m} \cdot p^{-(ps-1)} \leq N \leq (1 - p^{-m}) \cdot p^{(ps-1)}.$$

Приклад. При  $P = 2$ ,  $m = 22$  і  $s = 10$  діапазон чисел простягається приблизно від  $10^{-300}$  до  $10^{300}$ .

Зазначимо, що всі числа з плаваючою комою зберігаються в машині у т. зв. *нормалізованому вигляді*. Нормалізованим називають таке число, в старшому розряді мантиси якого стоїть одиниця. У нормалізованих двійкових чисел, отже,  $0,5 \leq M < 1$ .

**Алгебраїчне подання двійкових чисел.** Знак числа зазвичай кодують двійковою цифрою, при чому код 0 означає знак “+”, код 1 – знак “–”. Для алгебраїчного подання чисел у вичислювальних машинах використовують спеціальні коди: прямий код числа; обернений код числа; додатковий код числа, а два останні коди дають змогу замінити незручну для комп'ютера операцію віднімання операцією додавання з від'ємним числом.

Додатковий код забезпечує більш швидке виконання операцій, тому в комп'ютері використовують саме його.

1. *Прямий код числа*  $N - [N]_{\text{пр}}$ . Нехай:  $N = a_1, a_2, a_3 \dots a_n$ :

якщо  $N > 0$ , то  $[N]_{\text{пр}} = 0, a_1, a_2, a_3 \dots a_n$ ;

якщо  $N < 0$ , то  $[N]_{\text{пр}} = 1, a_1, a_2, a_3 \dots a_n$ ;

якщо  $N = 0$ , то має місце неоднозначність  $[0]_{\text{пр}} = 0, 0 \dots$  або  $= 1, 0 \dots$ .

Узагальнюючи результати, отримаємо:

$$[N]_{\text{пр}} = \begin{cases} N, & \text{якщо } N \geq 0 \\ 1 - N, & \text{якщо } N \leq 0 \end{cases}$$

У разі складання обидвох доданків, які мають однаковий знак операцію додавання виконують звичайним методом. Якщо ж доданки мають різні знаки, то спочатку необхідно виявити більше за абсолютною величиною число, виконати з нього віднімання меншого за абсолютною величиною і різниці присвоїти знак більшого числа.

Виконання операцій множення і ділення у прямому коді виконують звичайним чином, але знак результату визначають за збігом або незбігом знаків чисел, які беруть участь в операції.

Операцію віднімання у цьому коді не можна замінити операцією додавання з від'ємним числом, тому виникають складності, пов'язані з позикою значень зі старших розрядів зменшуваного числа. У зв'язку з цим прямий код в комп'ютері майже не використовують.

## 2. Оборнений код числа $N - [N]_{\text{об}}$ .

Позначення  $\bar{a}$  означає величину, зворотну  $a$  (інверсію  $a$ ), тобто якщо  $a=1$ , то  $\bar{a}=0$ , і навпаки:

якщо  $N > 0$ , то  $[N]_{\text{об}} = [N]_{\text{пр}} = 0, a_1, a_2, a_3 \dots a_n$ ;

якщо  $N < 0$ , то  $[N]_{\text{об}} = [N]_{\text{пр}} = 1, \bar{a}_1, \bar{a}_2, \bar{a}_3, \dots \bar{a}_n$ ;

якщо  $N=0$ , то має місце неоднозначність,  $[0]_{\text{об}} = 0,00 \dots 0$  або  $1,11 \dots 1$ .

Для того, щоб отримати оборнений код від'ємного числа, необхідно всі цифри цього числа інвертувати, тобто у знаковому розряді поставити 1, у всіх значущих розрядах нулі замінити одиницями, а одиниці нулями. Наприклад, число  $N=0,1011$ ,  $[N]_{\text{об}} = 10 - 1 \cdot 10^{-n} + N$ , тобто  $[N]_{\text{об}} = 1.1111 + N$ .

Узагальнюючи результати, отримаємо:

$$[N]_{\text{об}} = \begin{cases} N, & \text{якщо } N \geq 0 \\ 10 - 1 \cdot 10^{-m} + N, & \text{якщо } N \leq 0 \end{cases}$$

## 3. Додатковий код числа $N - [N]_{\text{дод}}$ .

якщо  $N \geq 0$ , то  $[N]_{\text{дод}} = [N]_{\text{пр}} = 0, a_1, a_2, a_3 \dots a_n$ ;

якщо  $N < 0$ , то  $[N]_{\text{дод}} = [N]_{\text{пр}} = 1, \bar{a}_1, \bar{a}_2, \bar{a}_3, \dots \bar{a}_n + 0,000 \dots 1$ .

Для того, щоб отримати додатковий код від'ємного числа, необхідно всі його цифри інвертувати (у знаковому розряді поставити одиницю; у всіх значущих розрядах нулі замінити одиницями, а одиниці – нулями) і потім до молодшого



розряду додати одиницю. У випадку потреби в перенесенні першого після коми розряду у знаковий розряд, до числа необхідно додати одиницю в молодший розряд.

Узагальнюючи, можна записати:

$$[N]_{\text{дод}} = \begin{cases} N, & \text{якщо } N \geq 0 \\ 10 + N, & \text{якщо } N \leq 0 \end{cases}.$$

*Особливості виконання операцій над числами з плаваючою комою (крапкою).* При додаванні (відніманні) чисел з однаковими порядками їх мантиси додають (віднімають) і результату присвоюють порядок, спільний для вихідних чисел. Якщо порядки вихідних чисел різні, то спочатку їх вирівнюють (число з меншим порядком приводиться до числа з більшим порядком), потім виконують операцію додавання (віднімання) порядків. Якщо при виконанні операції складання мантис виникає переповнення, то сума мантис зсувається вправо на один розряд, а порядок суми збільшується на одиницю.

У разі *множення* чисел з плаваючою комою їхні мантиси перемножують, а порядки додають.

При *діленні* чисел з плаваючою комою мантису діленого ділять на мантису дільника, а для отримання порядку частки з порядку діленого віднімають порядок дільника, при чому якщо мантиса діленого більша від мантиси дільника, то мантиса частки виявиться більше одиниці (переповнення) і її зсувають на один розряд вправо, водночас, збільшивши на одиницю порядок частки.

*Виконання арифметичних операцій над числами, поданими у додаткових кодах.* При виконанні арифметичних операцій у комп'ютері зазвичай застосовують не прості, а *модифіковані* коди. Модифікований код відрізняється від простого використанням для зображення знака числа двох розрядів. Другий знаковий розряд служить для автоматичного виявлення ситуації переповнення розрядної сітки: у разі браку переповнення обидва знакові розряди мають мати однакові цифри (нулі або одиниці), а при переповненні розрядної сітки цифри в них будуть різні. При переповненні результат зсувається вправо на один розряд.

Додають за звичайними правилами додавання двійкових чисел: одиницю перенесення, що виникає зі старшого знакового розряду, просто відкидають. Приклади додавання (кома умовно відокремлює знаковий розряд від самого числа): 1)  $X = -1101$ ,  $Y = 1001$ . Результат додавання:  $11,0011 + 00,1001 = 11,1100$  (або  $-0100$ ); 2)  $X = -1101$ ,  $Y = 1001$ . Результат :  $00,1101 + 00,1001 = 01,0110$  (переповнення, після зсуву вправо отримаємо  $01, 10110$  або  $+10110$ ); 3)  $X = 1101$ ,  $Y = -1001$ . Результат додавання  $00,1101 + 11, 0111 = 100,0100$  (або  $00,0100$ ); 4)  $X = -1101$ ,  $Y = -1001$ . Результат додавання  $11,0011 + 11,0111 = 10,1010$  (переповнення, після зсуву вправо отримаємо  $11,01010$  або  $-10110$ ).

Множення чисел у додаткових кодах виконують за звичайними правилами множення двійкових чисел. Єдиною особливістю є те, що якщо множник є від'ємним (знакові розряди рівні 11), то перед початком множення необхідно приписати до нього зліва стільки одиниць, скільки значущих розрядів є у другого співмножника справа від коми. Результат (добутку) завжди отримаємо у додатковому коді.

*Особливості подання інформації в ПК.* Числову інформацію всередині ПК кодують у двійковій або двійково-десятковій системах числення; при введенні – виведенні довільної інформації використовують спеціальні коди подання інформації – коди **A S C I I**. Ці ж коди застосовують для кодування буквенної і символної інформації усередині ПК. Для зручності роботи введені певні терміни для позначення сукупності двійкових розрядів (біт, байт, параграф, кбайт, мегабайт, гігабайт, терабайт, петабайт). Ці терміни звичайно використовують як одиниці вимірювання об'ємів інформації, оброблюваної комп'ютером. Послідовність декількох бітів або байтів називають *полем даних*. Біти в числі (в слові, в полі) нумерують справа наліво, починаючи з нульового розряду. В ПК можуть оброблюватися поля *постійної* і *змінної* величини. Поля постійної довжини: слово – 2 байти; подвійне слово – 4 байти; півслово – 1 байт; розширене слово – 8 байтів. Числа з фіксованою комою найчастіше мають формат слова і півслова; числа з плаваючою комою – формат подвійного і розширеного слова. Поля змінної довжини можуть мати довільний розмір від 0 до 255, але обов'язково рівний цілому числу байтів.

Двійково-кодовані десяткові числа можуть бути подані в ПК полями змінної довжини у т. зв. *упакованому* (див.: рис. 1.19) і *розпакованому* форматах. В упакованому форматі для кожної десяткової цифри відведено по чотири двійкові розряди (півбайта), при чому знак числа кодується в крайньому правому півбайті числа (1100 – знак “+” і 1101 – знак “-”).

ЦФ	ЦФ	ЦФ	...	ЦФ	Знак
					{Байт}

Рис. 1.19. Структура поля упакованого формату  
(ЦФ – цифра, знак–знак числа)

Упакований формат використовують в ПК звичайно при виконанні операцій складання і віднімання двійково-десяткових чисел.

У розпакованому форматі (див.: рис. 1.20) для кожної десяткової цифри виділяють по цілому байту, водночас старші півбайти (зона) кожного байта (крім самого молодшого) в ПК заповнюються кодом 0011 (відповідно до **ASCII** кодом), а в молодших (лівих) півбайтах звичайним чином кодуються десяткові

цифри. Старший півбайт (зона) самого молодшого (правого) байта використовують для кодування знака числа.

Зона	ЦФ	ЦФ	...	Зона	ЦФ	Знак	ЦФ
------	----	----	-----	------	----	------	----

Рис. 1.20. Структура поля розпакованого формату

Розпакований формат використовують у ПК при введенні–виведенні інформації, а також при виконанні операцій множення і ділення двійково-десяткових чисел.

Наприклад число  $-193_{(-10)} = -00011001001_{(2-10)}$  в ПК буде подане: в упакованому форматі:

0001	1001	0011	1101
------	------	------	------

у розпакованому форматі:

0011	0001	0011	1001	1101	0011
------	------	------	------	------	------

Код ASCII (американський стандартний код для обміну інформацією) має основний стандарт і його розширення. Основний стандарт для кодування символів використовує шістнадцятиричні коди 00-7F, розширення стандарту – 80-FF.

Основний стандарт є міжнародним, його застосовують для кодування управляючих символів, цифр, знаків пунктуації, букв латинського алфавіту та інших символів; у розширенні стандарту кодують символи псевдографіки і букви національного алфавіту.

### 1.6.2. Логічні основи комп'ютерної електроніки

**Основи алгебри логіки (булевої алгебри).** В засадах функціонування логічних пристроїв та вузлів комп'ютерної електроніки є *алгебра логіки* – науки, що використовує математичні методи для розв'язання логічних задач. Її використання ґрунтується на тому, що базові цифрові елементи функціонують у двох стійких станах і тому можуть бути описані саме такими математичними прийомами. Алгебру логіки називають *булевою алгеброю*, або *алгеброю висловлень*, початок якій покладено працями англійського математика Дж. Буля. *Булевою алгеброю* називають множину, що складається не менше, ніж з двох елементів, на якій визначені три операції – заперечення (НЕ), диз'юнкції (АБО), кон'юнкції (І).

Під висловленням розуміють будь-яке твердження, яке може бути *істинним* або *хибним*. Істинному висловленню приписують 1, хибному – 0. Висловлення можуть бути *простими* і *складними*. Складні висловлення складаються з про-

стих. Для об'єднання простих висловлень у складні використовують логічні зв'язки, що пов'язані логічними функціями, аргументами яких є прості висловлення. Для прикладу, прості висловлювання називають *аргументами* (змінними) і позначають літерами  $X_1, X_2 \dots X_n$ . У програмуванні та математиці аргумент або командно- оброблювальні дані називають *операндами*. Використовуючи логічні зв'язки **НІ, АБО, І, ЯКЩО** та ін. будують складні висловлювання, які отримали назву *булевих (логічних) функцій*. Їх позначають літерами  $F, L, K, M, P$  та ін.

**Операція** – детермінована дія над одним або декількома операндами, яка генерує новий результат (об'єкт). У булевих операціях операнди і результат набувають значення “1” і “0”. Булеву операцію над одним операндом називають *одномісною*, над двома – *двомісною* і т.д.

Загалом булеві функції можуть залежати від однієї, двох та  $n$  змінних. Зокрема для  $n$  змінних таку функцію має такий записують:  $F(X_1, X_2 \dots X_n)$ .

Областю визначення булевої функції  $F(X_1, X_2 \dots X_n)$  є скінчена множина різних двійкових наборів  $n$ , на кожному з яких вказують значення функції нуль або одиниці. Кількість різноманітних двійкових наборів дорівнює множині  $n$ –розрядних чисел  $m=2^n$ .

Основними булевими операціями є логічне заперечення (операція **НЕ**, інверсія), диз'юнкція (**АБО**, логічне додавання) і кон'юнкція (**І**, логічне множення).

**Логічний зв'язок “НЕ”** (заперечення) означає *заперечення висловлення* і читається “НЕ”. Булева функція *заперечення* – це одномісна булева функція  $F = \bar{X}$  (“не  $X$ ”) результатом якої є значення, протилежне значенню операнда.

**Диз'юнкцією** називають складне висловлення, що містить декілька простих висловлень і яке є істинним тоді, коли істинним буде хоч одне з простих висловлень, які входять в це складне висловлення, і хибним, якщо всі прості висловлення хибні. Диз'юнкція становить собою логічний зв'язок “АБО”, її позначають “ $\vee$ ”. Читається “або”. Булева функція *диз'юнкція* – це булева операція  $F = X_1 \vee X_2$  (“ $X_1$  або  $X_2$ ”), результатом якої є значення нуль тоді і тільки тоді, коли обидва операнди мають значення нуль.

**Кон'юнкцією** називають складне висловлення, що містить два або більше простих висловлень і яке є істинним тоді і лише тоді, коли істинними є прості висловлення, і хибним, якщо хоч одне з простих висловлень хибне. Кон'юнкція – це логічний зв'язок “І”. Читається “і”. Булева функція *кон'юнкція* – це булева операція  $F = X_1 \cdot X_2$  (“ $X_1$  і  $X_2$ ”) результатом якої є значення одиниці тоді і тільки тоді коли значення кожного операнда дорівнює одиниці. Іноді замість крапки застосовують знак “ $\wedge$ ” або “&”.

Описані попередньо операції можна задати за допомогою таблиць істинності. *Таблиця істинності* – це таблиця, що містить всі можливі комбінації вхідних логічних змінних і відповідні їм значення логічних функцій.

Розглянемо таблиці істинності базових логічних операцій.

Таблиця істинності функції **НЕ** сама проста і складається усього з двох рядків, оскільки вона має всього одну змінну, а кількість варіантів для єдиної булевої змінної рівне  $2^1 = 2$  (див.: табл. 1.1).

Таблиця 1.1

$X$	$F=X$
1	0
0	1

Таблиця 1.2

$X_1$	$X_2$	$F$
0	0	0
0	1	1
1	0	1
1	1	1

Таблиця 1.3

$X_1$	$X_2$	$X_3$	$F$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Таблиця істинності функції **АБО** для двох змінних складається з чотирьох рядків (див.: табл. 1.2). Кількість поєднань цих змінних рівне  $2^2 = 4$ .

Таблиця істинності логічного множення (кон'юнкції) – функції **І** для трьох логічних змінних подана в таблиці 1.3. Кожна з цих змінних може бути або у стані логічного 0 або логічної одиниці 1. Отже, кількість поєднань цих змінних рівна  $2^3 = 8$ .

Розглянуті логічні функції можна змоделювати за допомогою простих електричних схем, що містять ключі ( $A$ ,  $B$ ,  $C$ ) та які можуть перебувати у розімкнутому ( $A=0$ ) або замкнутому ( $A=1$ ) станах. Тоді за логічний 0 на виході схеми ( $F=0$ ) вважають такий її стан, у якому через опір  $R$  струм не протікає.

Протікання струму означатиме значення функції  $F=1$ .

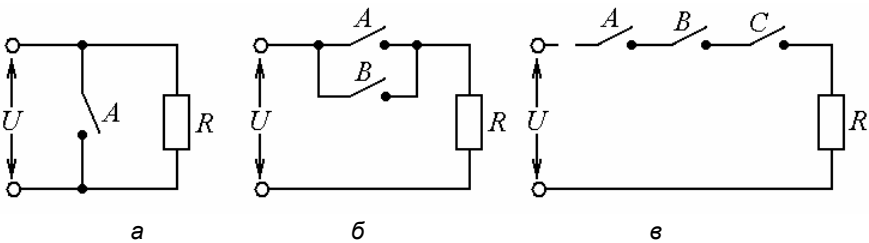


Рис.1.21. Схеми реалізації логічних функцій НЕ (а), АБО (б) та І (в)

На рисунку 1.21, а зображена схема, що реалізує логічну функцію **НЕ**. Очевидно, що через опір  $R$  струм протікає ( $F=1$ ), коли  $A$  не замкнений, тобто  $A=0$ .

Логічну функцію **АБО** реалізує схема, подана на рисунку 1.21, б. Очевидно, що через опір  $R$  струм протікає тоді, коли замкнуті **або** ключ  $A$ , **або**  $B$ . Звідси і назва логічного додавання.

У схемі на рисунку 1.21, в, що реалізує логічну функцію **І** (кон'юнкція), струм через опір  $R$  протікає лише тоді, коли замкнуті і  $A$ , і  $B$ , і  $C$ . Тому такий логічний елемент назвали елементом **І**.

Отже для отримання в комп'ютерній електроніці логічних 0, 1 необхідно реалізувати стан певних схем з детермінованими параметрами. Для прикладу, в сучасній схемотехніці для логічних елементів і схем, виконаних за технологією транзисторно-транзисторної логіки (ТТЛ-схеми), логічний 0 реалізується напругою в діапазоні  $0 \dots +0,4$  В, а логічна 1 – у діапазоні  $+2,4 \dots +5$  В.

Для основних булевих операцій справджуються такі закони, тотожності, властивості:

- 1) асоціативність (сполучний закон):  $X_1 \vee X_2 X_3 = (X_1 X_2 X_3; X_1 X_2 X_3) = (X_1 X_2) X_3$ ;
- 2) комутативність (переміщувальний закон):  $X_1 X_2 = X_2 X_1$ ;  $X_1 X_2 = X_2 X_1$ ;
- 3) дистрибутивність (розподільний закон):  $X_1 (X_2 \vee X_3) = X_1 X_2 \vee X_1 X_3$ ;  
 $X_1 \vee X_2 X_3 = (X_1 \vee X_2)(X_1 \vee X_3)$ ;
- 4) ідемпотентність (виключення повторення):  $X \vee X \vee X \vee X = X$ ;  $X X X = X$ ;
- 5) закон поглинання:  $X_1 \vee X_1 X_2 = X_1$ ;  $X_1 (X_1 \vee X_2) = X_1$ ;
- 6) закон склеювання:  $X_1 X_2 X_1 \bar{X}_2 = X_1$ ;  $(X_1 \vee X_2)(X_1 \vee \bar{X}_2) = X_1$ ;
- 7) закон де-Моргана:  $\overline{X_1 \vee X_2} = \bar{X}_1 \bar{X}_2$ ;  $\overline{X_1 X_2} = \bar{X}_1 \vee \bar{X}_2$ ;
- 8) властивості заперечення і констант:

$$X \vee \bar{X} = 1; \quad X \bar{X} = 0; \quad \bar{\bar{X}} = X; \quad \bar{1} = 0; \quad \bar{0} = 1;$$

$$X \vee 0 = X; \quad X \vee 1 = 1; \quad X \cdot 1 = X; \quad X \cdot 0 = 0;$$

- 9) тотожності:  $X_1 \vee \bar{X}_1 X_2 = X_1 \vee X_2$ ;  $X_1 (X_1 \vee X_2) = X_1 X_2$ .

Виконання всіх цих законів, властивостей тотожностей перевіряють відставляючи в логічний вираз змінні нуля та одиниці.

Запропоновані такі нові булеві функції:

**виключення (заборона)** – двомісна булева операція, результатом якої є значення одиниця тоді і тільки тоді, коли значення одного операнда дорівнює одиниці, а іншого – нулю:

$$F_2 = X_1 \bar{X}_2, \text{ або } F_2 = \bar{X}_1 X_2.$$

**Сума за модулем два** (виключальне АБО, заперечення еквівалентності) – двомісна булева операція, результатом якої є значення одиниці тоді і тільки тоді, коли операнди мають різні значення. Позначають так:

$$F_6 = X_1 + X_2 = \bar{X}_1 X_2 = \bar{X}_1 X_2 + X_1 \bar{X}_2.$$

**Заперечення диз'юнкції** (операція НЕ, АБО, стрілка Пірса) – булева операція, результатом якої є значення одиниці тоді і тільки тоді, коли обидва операнди дорівнюють нулю. Позначають так:

$$F_8 = X_1 \downarrow X_2 = \overline{X_1 \vee X_2}.$$

**Еквівалентність** (рівнозначність) – двомісна булева операція, результатом якої є одиниця тоді і тільки тоді, коли операнди набувають однакових значень. Позначають у вигляді:

$$F_9 = X_1 \sim X_2 = X_1 X_2 \vee \bar{X}_1 \bar{X}_2.$$

Значення булевих функцій двох змінних подані в таблиці 1.4.

Таблиця 1.4

Булеві функції двох змінних

Номер за пор.	Значення булевих функцій залежно від аргументів $X_1, X_2$					Позначення функції	Назви функції
1	2					3	4
	$X_1$	0	0	1	1		
	$X_2$	0	1	0	1		
1	$F_0(X_1, X_2)$	0	0	0	0	$F_0 = 0$	Константа 0
2	$F_1(X_1, X_2)$	0	0	0	1	$F_1 = X_1 \wedge X_2$	Кон'юнкція, логічне множення
3	$F_2(X_1, X_2)$	0	0	1	0	$F_2 = X_1 \bar{X}_2$	Заборона по $X_2$ , заперечення імплікації
4	$F_3(X_1, X_2)$	0	0	1	1	$F_3 = X_1$	Повторення $X_1$ ; змінна $X_1$
5	$F_4(X_1, X_2)$	0	1	0	0	$F_4 = \bar{X}_1 X_2$	Заборона по $X_1$ , заперечення імплікації
6	$F_5(X_1, X_2)$	0	1	0	1	$F_5 = X_2$	Повторення $X_2$ ; змінна $X_2$
7	$F_6(X_1, X_2)$	0	1	1	0	$F_6 = X_1 \oplus X_2$	Сума за модулем 2, логічна нерівнозначність

Закінчення табл. 1.4

8	$F_7(X_1, X_2)$	0	1	1	1	$F_7 = X_1 \vee X_2$	Диз'юнкція
9	$F_8(X_1, X_2)$	1	0	0	0	$F_8 = X_1 \downarrow X_2$	Заперечення диз'юнкції стрілка Пірса
10	$F_9(X_1, X_2)$	1	0	0	1	$F_9 = X_1 \sim X_2$	Еквівалентність
11	$F_{10}(X_1, X_2)$	1	0	1	0	$F_{10} = \bar{X}_2$	Заперечення $X_2$ , інверсія $X_2$
12	$F_{11}(X_1, X_2)$	1	0	1	1	$F_{11} = X_1 \leftarrow X_2$	Імплікація від $X_2$ до $X_1$
13	$F_{12}(X_1, X_2)$	1	1	0	0	$F_{12} = \bar{X}_1$	Заперечення $X_1$ , інверсія $X_1$
14	$F_{13}(X_1, X_2)$	1	1	0	1	$F_{11} = X_1 \rightarrow X_2$	Імплікація від $X_1$ до $X_2$
15	$F_{14}(X_1, X_2)$	1	1	1	0	$F_{11} = X_1 / X_2$	Штрих Шеффера, заперечення кон'юнкції
16	$F_{15}(X_1, X_2)$	1	1	1	1	$F_{15} = 1$	Константа 1

**Імплікація** (включення) – двомісна булева операція, результатом якої є значення нуль тоді і тільки тоді, коли операнди набувають однакових значень. Позначають так:

$$F_{11} = X_1 \leftarrow X_2 = X_1 \bar{X}_2;$$

$$F_{13} = X_1 \rightarrow X_2 = \bar{X}_1 X_2.$$

**Заперечення кон'юнкції** (операція НЕ І, штрих Шеффера, заперечення перетину) – булева операція, результат якої дорівнює нулю тоді і тільки тоді, коли обидва операнди дорівнюють одиниці. Позначають у вигляді:

$$F_{14} = X_1 / X_2 = \bar{X}_1 \bar{X}_2.$$

Схемну реалізацію певної логічної операції здійснюють у комп'ютерній електроніці *логічним елементом (вентилем)*. Набір взаємозалежних логічних елементів у поєднанні з формальними методам взаємозв'язку називають *логічною схемою*.

Назви та вітчизняні і зарубіжні позначення головних логічних елементів, які застосовують у комп'ютерній електроніці зображені у таблиці 1.5.



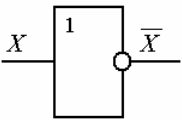
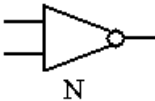
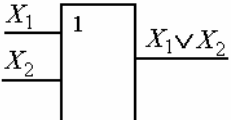
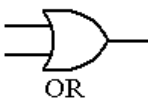
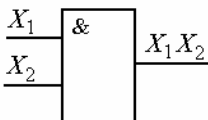
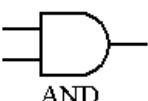
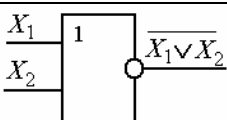
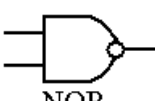
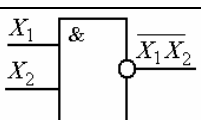
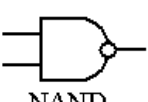
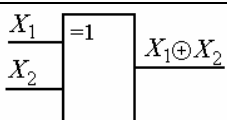

**Задавання логічних функцій.** Для опису функцій алгебри логіки можуть бути використані різні способи. Основними з них є: опис у словесній формі, у вигляді таблиць істинності, алгебраїчних виразів, послідовності десяткових чисел, часовими діаграмами, геометричними фігурами та графами.

**Словесний опис функцій алгебри логіки.** Такий спосіб найбільш часто застосовують для первинного, початкового опису поведінки логічного пристрою. *Приклад:* логічна функція трьох змінних рівна одиниці, якщо хоча б дві вхідні змінні рівні одиниці.

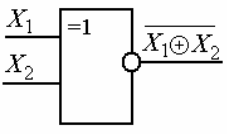

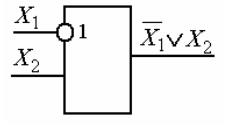
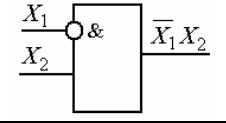
**Опис функцій алгебри логіки у вигляді таблиць істинності.** Таблицю, яка містить всі можливі комбінації вхідних змінних  $X_{n-1}, \dots, X_1, X_0$  і значення вихідних змінних  $F_i$ , називають *таблицею істинності* або *комбінаційною таблицею*. Узгалянено таблиця істинності містить  $2^n$  рядки.

Таблиця 1.5

Позначення головних логічних елементів

Назва операції	Назва елемента	Умове графічне позначення	ANSI
1	2	3	4
Заперечення	НЕ		
Диз'юнкція	АБО		
Кон'юнкція	І		
Заперечення диз'юнкції	АБО-НЕ		
Заперечення кон'юнкції	І-НЕ		
Заперечення еквівалентності	Виключальне АБО		

Закінчення табл. 1.5

1	2	3	4
Еквівалентність	Еквівалентність		
Імплікація	Якщо, ТО		—
Заборона	НЕ		—

**Опис функцій алгебри логіки у вигляді алгебраїчного виразу.** Алгебра логіки дає змогу створювати складні функції, аргументи яких є функції інших двійкових аргументів. Операція заміни аргументу однієї функції іншими, більш простими функціями має назву *суперпозиції функцій*. Багатократне використання принципу суперпозиції дає можливість отримати функції бажаного числа аргументів.

Елементарну кон'юнкцію отримують кон'юнкцією скінченної множини логічних змінних та їхніх інверсій.

Приклад:  $P(X_1, X_2, X_3) = X_1 X_2 \bar{X}_3$ .

Елементарну диз'юнкцію отримуєть диз'юнкцією скінченної множини логічних змінних та їхніх інверсій.

Приклад:  $P(X_1, X_2, X_3) = X_1 + X_2 + \bar{X}_3$ .

Кількість змінних в елементарній кон'юнкції (диз'юнкції) називають її довжиною, що визначає її ранг.

Приклад:  $P(X_1, X_2, X_3, X_4) = X_1 + X_2 + \bar{X}_3 + X_4$  є диз'юнкція четвертого рангу.

*Мінтермом* називають функцію, що приймає одиничне значення на одному зі всіх можливих наборів аргументів, а *макстермом* – функцію, яка приймає нульове значення за одного з можливих наборів і одиничне значення за всіх інших. Мінтерм алгебраїчно становить собою кон'юнкцію аргументів, а макстерм – диз'юнкцію аргументів. Якщо використовують двійкову систему і кількість наборів аргументів  $n$ , то кількість мінтермів або макстермів  $N=2^n$ .

Диз'юнкцію довільного числа елементарних кон'юнкцій називають *диз'юнктивною нормальною формою* (ДНФ).

Приклад:  $X_1 + X_2 X_3 + \bar{X}_1 X_2 X_3 + X_1 \bar{X}_2 \bar{X}_3$ .

Кон'юнкцію довільного числа елементарних диз'юнкцій називають *кон'юнктивною нормальною формою* (КНФ).

Приклад:  $X_1 (X_1 + X_2) (\bar{X}_2 + X_3) (\bar{X}_1 + X_2 + \bar{X}_3)$ .

Нормальні форми логічних функцій прийнято називати *канонічними*. Логічну функцію, задану довільним аналітичним виразом, можна безпосередньо перетворити у нормальну диз'юнктивну (або кон'юнктивну) форми. Для цього необхідно: виразити всі операції через операції кон'юнкції, диз'юнкції та інверсій; позбавитися від інверсії над цілими виразами, перейшовши до форми, в якій є інверсія лише окремих змінних; розкрити дужки, застосовуючи закон дистрибутивності; привести кон'юнкції (диз'юнкції) до елементарних.

Якщо до складу логічної формули входять набори елементарних кон'юнкцій однакового рангу, пов'язані з диз'юнкцією, то така форма подання логічної функції має назву *досконалої диз'юнктивної нормальної форми* (ДДНФ). Правило утворення ДДНФ функції  $n$  аргументів полягає в такому: 1) за кожним набором двійкових змінних, за якого функція набуває значення одиниці, скласти елементарні кон'юнкції (мінтерми); 2) в елементарну кон'юнкцію записати не інвертованими змінні, задані одиницею в таблиці істинності, а інвертованими ті змінні, які в таблиці істинності задані нулем. Отримані з'єднання називають *конституєнтами одиниці*; 3) елементарні кон'юнкції з'єднати знаком диз'юнкції.

*Досконалою нормальною кон'юнктивною формою* (ДНКФ) логічної функції прийнято називати такий її вираз, який містить елементарні диз'юнкції одного рангу, пов'язані кон'юнкцією. Правило утворення ДНКФ  $n$  аргументів полягає у такому: 1) з кожного набору двійкових змінних, за якого функція має значення нуля, скласти елементарні диз'юнкції (макстерми); 2) в елементарні диз'юнкції записати не інвертованими змінні, задані нулем в таблиці істинності, а інвертованими ті змінні, які – одиницею. Отримані суми називають *конституєнтами нуля*; 3) елементарні диз'юнкції з'єднати знаком кон'юнкції.

Приклад: нехай таблицею істинності задана функція  $F(X_2, X_1, X_0)$  (див. табл. 1.6). Потрібно перетворити її в ДДНФ і ДНКФ.

За таблицею знаходимо, що функція  $F$  набуває значення одиниці за чотирьох наборів аргументів, тому функція ДДНФ буде складатися з логічної суми чотирьох мінтермів:

$$F(X_2, X_1, X_0) = \bar{X}_2 \bar{X}_1 X_0 + \bar{X}_2 X_1 \bar{X}_0 + X_2 \bar{X}_1 \bar{X}_0 + X_2 X_1 X_0.$$

Функція  $F(X_2, X_1, X_0)$  в ДНКФ набуває значення нуля за чотирьох наборів аргументів, вона буде складатися з логічного поєднання чотирьох макстермів:

$$F(X_2, X_1, X_0) = (X_2 + X_1 + X_0) (X_2 + \bar{X}_1 + \bar{X}_0) (\bar{X}_2 + X_1 + \bar{X}_0) (\bar{X}_2 + \bar{X}_1 + X_0).$$

Таблиця 1.6

Таблиця істинності функції  $F(X_2, X_1, X_0)$ 

Значення аргумента			Значення функції $F$	ДДНФ	ДКНФ
$X_2$	$X_1$	$X_0$		мінтерм	макстерм
0	0	0	0	–	$X_2 + X_1 + X_0$
0	0	1	1	$\bar{X}_2 \bar{X}_1 X_0$	–
0	1	0	1	$\bar{X}_2 X_1 \bar{X}_0$	–
0	1	1	0	–	$X_2 + \bar{X}_1 + \bar{X}_0$
1	0	0	1	$X_2 \bar{X}_1 \bar{X}_0$	–
1	0	1	0	–	$\bar{X}_2 + X_1 + \bar{X}_0$
1	1	0	0	–	$\bar{X}_2 + \bar{X}_1 + X_0$
1	1	1	1	$X_2 X_1 X_0$	–

ДКНФ використовують рідше, ніж ДДНФ у процесі перетворення логічних виразів. Отримані формули функції, якщо не застосовувати перетворень, можна використати для синтезу функціональних схем логічних пристроїв.

Від довільної ДНФ можна перейти до ДДНФ функції за допомогою рівносильних перетворень. Такий перехід називають *розгортанням*. Для цього необхідно: 1) ввести змінні у кожен кон'юнкційний множитель на рівнозначність виду  $X_i + \bar{X}_i = 1$ , де  $X_i$  – відсутня змінна; 2) розкрити дужки, застосовуючи комутативний закон; 3) позбавитися від кон'юнкцій, які повторюються на основі закону ідемпотентності.

*Опис функцій алгебри логіки у вигляді послідовності десяткових чисел.* З метою скорочення запису функцію алгебри логіки іноді подають у вигляді послідовності десяткових чисел. Водночас послідовно записують десяткові еквіваленти двійкових кодів конститuent одиниці або нуля.

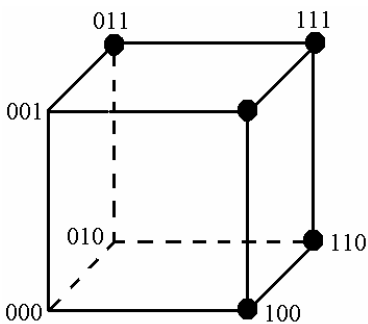


Рис. 1.22. Геометричне подання функції  $F(X_2, X_1, X_0) = \sum (3, 4, 5, 6, 7)$

*Кубічні комплекси.* Основою кубічної форми є подання кожного набору вхідних змінних у вигляді  $n$ -мірного вектора. Вершини цих векторів геометрично можуть бути подані як вершини  $n$ -мірного куба. Відзначаючи точками вершини векторів, для яких функція рівна одиниці, отримаємо геометричне подання функції у вигляді куба.

Приклад: задана функція  $F(X_2, X_1, X_0) = \sum (3, 4, 5, 6, 7)$ . Геометрично подати у вигляді куба. Графічний розв'язок задачі проілюстровано на рисунку 1.22.

Набори змінних розміщені на кінцях ребер куба, відрізняються лише однією змінною. Їх прийнято називати *сусідніми*. Кожну вершину, в якій функція набуває одиничне значення, називають *нульовим кубом* (0-кубом). Записують 0-куб послідовністю вхідних змінних, які утворили його, тобто кодом, що відповідає конститuantі одиниці. Множина нульових кубів утворює нульовий кубічний комплекс  $K_0$ .

Якщо два нульові куби комплексу  $K_0$  відрізняються лише за однією координатою (змінною), тобто два набори змінних, для яких функція рівна одиниці, є сусідніми, то вони утворюють одиничний куб (1-куб). Геометрично це відповідає ребру початкового  $n$ -мірного куба. Записують 1-куб послідовністю спільних елементів його 0-кубів з прочерком елементів, що не збігаються. Множина одиничних кубів утворює одиничний кубічний комплекс  $K_1$ .

Аналогічно, якщо два одиничні куби комплексу  $K_1$  відрізняються лише за однією координатою (змінною), то вони утворюють двійковий куб (2-куб). Геометрично це відповідає грані початкового  $n$ -мірного куба.

Розмірність куба (його ранг) визначається кількістю координат, що не співпадають. Об'єднання кубічних комплексів  $K_0, K_1, \dots, K_m$  для функції логічної алгебри  $n$ -змінних утворює її кубічний комплекс  $K(z) = (K_0, K_1, \dots, K_m) \dots$ .

*Часова діаграма.* Одним з популярних способів задавання булевих функцій є використання часових діаграм. Значення змінних кодують за наявністю або браку імпульсів ( $X_1, X_2$ ) у певному часовому інтервалі. Значення булевої функції відображається графічно згідно з законами булевої алгебри (див.: рис. 1.23).

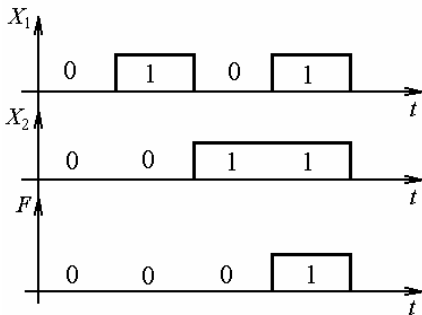


Рис. 1.23. Приклад задавання булевої функції за допомогою часової діаграми

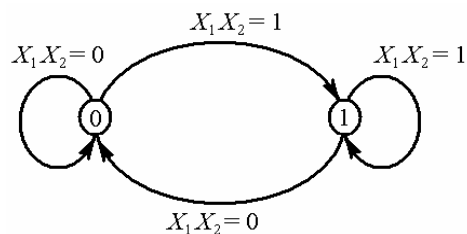


Рис. 1.24. Приклад задавання логічної функції за допомогою графа

*Задавання логічних функцій графами.* На рисунку 1.24 є приклад опису булевої функції графом (див.: рис. 1.24).

Тут вершини відображають значення нуля і одиниці, а на орієнтованих дугах змінні вказують умови переходів.

**Контрольні запитання до розділу 1**

1. Дайте означення науки "комп'ютерна електроніка" та сформулюйте її основні завдання.
2. Охарактеризуйте основні етапи розвитку засобів інформаційного спілкування, відомі сьогодні людству.
3. Які основні результати "нульового" покоління розвитку комп'ютерної електроніки ?
4. Охарактеризуйте перше покоління розвитку комп'ютерної електроніки.
5. Що є характерним для другого покоління розвитку комп'ютерної електроніки ?
6. Хто і коли створив перший мікропроцесор?
7. Порівняйте четверте і п'яте покоління розвитку комп'ютерної електроніки.
8. Охарактеризуйте вплив розвитку мікроелектроніки на продуктивність комп'ютерних систем.
9. Як впливає геометричний скейлінг у мікроелектроніці на основні параметри елементної бази комп'ютерної електроніки?
10. Дайте визначення поняття "інформація".
11. Опишіть узагальнену схему процесу передавання інформації.
12. Сформулюйте властивості інформації.
13. Які є методи оцінення кількості інформації ?
14. Що таке ентропія повідомлень та які вона має властивості?
15. Сутність методу оцінення кількості інформації за Шенноном.
16. Дайте означення поняттю "сигнал".
17. Яка класифікація сигналів є загальноприйнята ?
18. Які способи представлення сигналів Вам відомі?
19. Різновиди сигналів, які описують часовою функцією.
20. Які інформаційні параметри має часо-імпульсне подання прямокутних імпульсів?
21. Розкрийте сутність аналогового та цифрового подання інформації.
22. Які є види модуляції аналогових сигналів?
23. Чому завадостійкість цифрових сигналів вища, ніж аналогових?
24. Нарисуйте блок-схему ПК.
25. Дайте характеристику основних блоків комп'ютера.
26. Коротко охарактеризуйте пристрої, що входять до складу МП.
27. Що таке системна шина?
28. Яка основна характеристика системної шини ?
29. Наведіть ієрархію ЗП ПК.
30. Поясніть призначення ЗП ПК.
31. Класифікуйте зовнішні пристрої ПК.
32. Назвіть склад пристроїв кожної групи зовнішніх пристроїв ПК.
33. Що таке математичний співпроцесор, яке його призначення?
34. Що таке контролер прямого доступу до пам'яті (DMA), яке його призначення?
35. Що таке контролер переривань і яке його призначення?

36. Назвіть основні конструктивні компоненти ПК і дайте їм коротку характеристику.
37. Назвіть і поясніть основні функціональні характеристики ПК.
38. Чим визначається продуктивність комп'ютера?
39. Що таке системи числення та які їхні різновиди?
40. Запишіть довільне змішане число у системі числення з основою  $P$  у вигляді полінома.
41. Сформулюйте алгоритм перетворення чисел з десятикової системи у систему з основою  $P$ .
42. Сутність подання чисел з фіксованою та плаваючою комами.
43. Запишіть умови формування прямого та оберненого кодів числа.
44. Яка структура поля упакованого формату?
45. Сформулюйте означення булевої алгебри.
46. Типи висловлювань. Операнди.
47. Базові логічні функції.
48. Закони властивості та тотожності, що справджуються для булевих операцій.
49. Нові булеві функції.
50. Способи задавання логічних функцій.