

Г. Г. ЗЛОБІН, Р. Є. РИКАЛЮК

# АРХІТЕКТУРА ТА АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ ПЕОМ

ВИДАВНИЦТВО “КАРАВЕЛА”

Г. Г. Злобін, Р. Є. Рикалюк

# АРХІТЕКТУРА ТА АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ ПЕОМ

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

Київ “Каравела” 2006

УДК 004.7(075.8)  
ББК 32.973я73  
З 14

*Гриф надано  
Міністерством освіти і науки України  
(лист № 14/18.2-115 від 20.01.2005 р.)*

**Автори:**

**Злобін Г. Г.,**

кандидат технічних наук, доцент

Львівського національного університету ім. І. Франка;

**Рикалюк Р. Є.,**

кандидат фізико-математичних наук, доцент,

директор інформаційно-обчислювального центру

Львівського національного університету ім. І. Франка.

**Рецензенти:**

**Лізунов П. П.,**

доктор технічних наук, професор, завідувач кафедри основ інформатики  
Київського національного університету будівництва і архітектури;

**Чаплига В. М.,**

доктор технічних наук, професор, завідувач кафедри  
економічної кібернетики Львівського банківського інституту.

**Злобін Г. Г., Рикалюк Р. Є.**

**З 14 Архітектура та апаратне забезпечення ПЕОМ:** Навч. посіб. – К.:  
Каравела, 2006. – 304 с.

**ISBN 966-8019-41-5**

У навчальному посібнику викладені історія розвитку комп'ютерної техніки, основи алгебри логіки, головна елементна база сучасних ПЕОМ. Розглянуто принципи роботи сучасних процесорів, запам'ятовуючих та периферійних пристроїв, систему шин, базову систему вводу/виводу. Подано інформацію про апаратне забезпечення ПЕОМ: відеосистему, дискову підсистему, обладнання для локальних та глобальних мереж та засоби Multimedia.

Посібник призначений для студентів природничих і технічних факультетів вищих навчальних закладів та всіх, хто цікавиться інформатикою та обчислювальною технікою.

**УДК 004.7(075.8)  
ББК 32.973я73**

**ISBN 966-8019-41-5**

© Злобін Г. Г., Рикалюк Р. Є., 2006  
© Видавництво "Каравела", 2006

## Зміст

Вступ .....	6
<b>Розділ I. ТЕОРЕТИЧНІ ОСНОВИ ФУНКЦІОНУВАННЯ ЕОМ .....</b>	<b>19</b>
1.1. Поняття про інформацію .....	19
1.2. Основи алгебри логіки .....	21
1.3. Елементна база ЕОМ .....	26
Головні логічні елементи та вузли .....	27
RS-тригери з роздільним установленням 0 та 1 .....	28
D-тригер (затримки) .....	29
JK-тригер (універсальний) .....	29
T-тригер (лічильний) .....	30
1.4. Класифікація архітектур ОС .....	40
1.4.1. Класифікація за інтегральними ознаками .....	40
1.4.1.1. Класифікація за взаємодією ЦП, ОП, ПП .....	40
1.4.1.2. Класифікація за взаємодією потоку команд і потоку даних (за М.Флінном) .....	44
1.4.2. Класифікація за функціональним призначенням .....	47
1.4.3. Класифікація за способом організації виконання команд .....	47
1.5. Типи даних ПЕОМ .....	50
<b>Розділ II. ЗАГАЛЬНІ ПРИНЦИПИ ПОБУДОВИ ПЕОМ .....</b>	<b>54</b>
2.1. IBM-подібні ПЕОМ .....	55
2.1.1. Програмна архітектура IBM-подібних ПЕОМ .....	55
2.1.1.1. Регістрова структура 16-розрядних МП фірми Intel .....	55
2.1.1.2. Регістрова структура 32-розрядних МП фірми Intel .....	58
2.1.1.3. Переривання роботи мікропроцесора .....	68
Типи переривань і винятків 32-розрядних МП .....	72
Опрацювання переривань .....	74
Масковані переривання .....	74
Немасковані переривання .....	74
Програмні переривання .....	75
Пріоритетність переривань .....	75
2.1.1.4. Організація пам'яті та адресування .....	76
а) 16-розрядні процесори .....	76
Організація сегмента .....	76
Режими адресування .....	78
б) 32-розрядні процесори. Захищений режим. Механізми адресування .....	85
Поняття про адресні простори .....	86
Використання сегментного регістра .....	88
Простір введення/виведення .....	89
Сегментування .....	89
Дескрипторні таблиці .....	90
Опис дескрипторів .....	92
Посторінкова організація пам'яті .....	98



Елементи таблиці і вказівник сторінок .....	100
Режими адресування .....	100
Регістровий і безпосередній режими .....	100
Інші режими адресування пам'яті .....	101
Адресування пам'яті в реальному режимі .....	103
Запасні комірки пам'яті .....	104
2.1.1.5. Захист та привілеї .....	105
Правила привілеїв .....	106
Рівні привілеїв .....	106
2.1.1.6. Кешування пам'яті у персональних комп'ютерах .....	109
2.1.1.7. Віртуальна конфігурація 8086 .....	113
Механізм адресування у віртуальному 8086 режимі .....	114
2.1.1.8. Тестування. Самоконтроль .....	115
Тестування буфера асоціативної трансляції .....	115
Забезпечення налагодження .....	116
2.1.1.9. Режим системного керування SMM .....	118
2.1.1.10. Програмна архітектура співпроцесора .....	120
2.1.2. Апаратна реалізація головних вузлів ПЕОМ .....	122
Системна плата .....	122
2.1.2.1. Процесор, співпроцесор .....	125
2.1.2.2. Мікропроцесор Intel 8086 .....	127
2.1.2.3. Мікропроцесор 180286 .....	129
2.1.2.4. Мікропроцесор 180386 .....	132
2.1.2.5. Мікропроцесор 180486 .....	134
2.1.2.6. Мікропроцесор Pentium I .....	136
2.1.2.7. Мікропроцесори Pentium Pro і Pentium II .....	139
Про розширення MMX .....	140
2.1.2.8. Мікропроцесор Pentium III .....	141
2.1.2.9. Мікропроцесор Pentium IV .....	142
2.1.2.10. Intel-сумісні процесори інших виробників .....	145
2.1.2.11. 64-бітні мікропроцесори .....	150
2.1.2.12. Процесори Power та PowerPC .....	151
2.1.3. Оперативна пам'ять .....	154
2.1.4. Кеш-пам'ять .....	160
2.1.5. Система шин .....	162
2.1.6. Базова система вводу-виводу (BIOS) .....	165
2.1.7. Програма POST самотестування ПЕОМ .....	165
2.1.8. Програма Setup задання параметрів обладнання .....	167
2.1.9. Методи запису і модернізації BIOS .....	170
2.1.10. Системні засоби вводу-виводу .....	171
2.1.10.1. Клавіатура .....	171
2.1.10.2. Координатно-вказівні пристрої (mouse, track-boll, touch-pad) .....	174
2.1.10.3. Ігровий контролер .....	176
2.1.10.4. Паралельні інтерфейси .....	177

2.1.10.5. Послідовні інтерфейси .....	180
2.1.10.6. Бездротові інтерфейси .....	185
2.2. Особливості ПЕОМ фірми Apple .....	186
2.3. Особливості портативних ПЕОМ .....	192
2.3.1. Переносні ПЕОМ .....	192
2.3.2. Кишенькові ПЕОМ .....	194
2.3.3. Планшетні ПЕОМ .....	198
2.3.4. Комунікатори .....	199
2.4. Робочі станції і промислові ПЕОМ .....	199
2.5. Кластери .....	203
2.6. Серверні ЕОМ .....	205
2.7. Живлення та електробезпека ПЕОМ .....	208
Розділ III. АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ ПЕОМ .....	215
3.1. Відеосистема .....	215
3.1.1. Дисплеї і відеопроєктори .....	215
3.1.2. Графічні контролери .....	220
3.2. Дискова підсистема .....	225
3.2.1. Принципи магнітного запису .....	225
Магнітні диски .....	227
3.2.2. Оптичні диски .....	237
3.2.3. Пристрої архівування інформації .....	244
3.2.4. Пристрої на базі флеш-пам'яті .....	245
3.3. Інші пристрої вводу-виводу .....	252
3.3.1. Пристрої друку .....	252
3.3.2. Сканери .....	257
3.3.3. Цифрові фотокамери .....	259
3.4. Обладнання для локальних та глобальних мереж .....	263
3.4.1. Загальні принципи побудови обчислювальних мереж .....	264
3.4.2. Пристрої для побудови локальних мереж .....	269
3.4.3. Бездротові мережі .....	271
3.4.3.1. Бездротові персональні мережі .....	271
3.4.3.2. Бездротові локальні мережі .....	273
3.4.4. Пристрої для роботи у глобальних мережах .....	277
3.4.5. Обладнання для IP-телефонії .....	283
3.5. Засоби Multimedia .....	286
3.5.1. Звукові карти .....	286
3.5.2. Засоби запису і прослуховування звуку .....	287
3.5.3. Системи віртуальної реальності .....	288
3.5.4. Web-камери .....	289
3.5.5. Приймачі FM- та TV-сигналів .....	290
3.6. Встановлення ПЕОМ .....	291
СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ .....	293
Додаток 1. МЕТОДИ ПОШУКУ НЕСПРАВНОСТЕЙ У ПЕОМ .....	295
Додаток 2. МОДЕРНІЗАЦІЯ ПЕОМ .....	300

## ВСТУП

Електронно-обчислювальні машини (ЕОМ) з'явилися у 40-х рр. ХХ ст. Однак пристрої для проведення обчислень люди почали конструювати значно раніше. Зовсім недавно дослідники знайшли у щоденниках Леонардо да Вінчі (1452-1519) ескіз сумуючої машини на зубчастих колесах, спроможної додавати 18-розрядні десяткові числа. Спеціалісти ІВМ відтворили цей ескіз у металі і машина виявилася працездатною!

На початку ХVІІ ст. німецький вчений Вільгельм Шікард (1592-1636) у листі Йоганну Кеплеру подає малюнок лічильної машини для сумування і множення 6-розрядних десяткових чисел. Про це також стало відомо у наші дні. У 1617 р. Джон Непер розробив дерев'яний арифмометр. Для полегшення процесу обчислення податків Блез Паскаль у 1642 р. створив механічний пристрій для виконання операцій додавання і віднімання (спочатку 6 десяткових розрядів, згодом 8 десяткових розрядів). Було створено більше 50 зразків цієї машини.

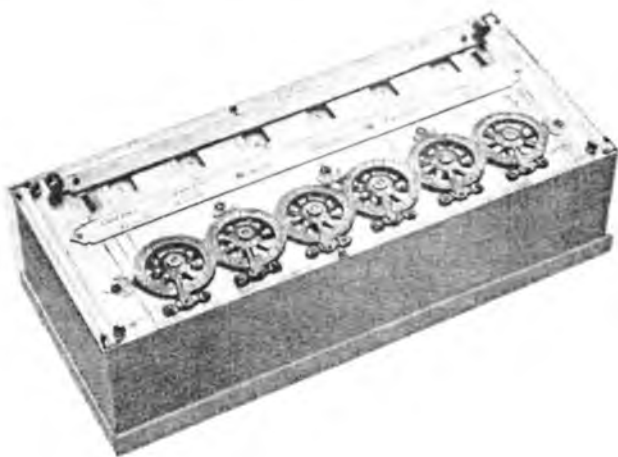


Рис. 1. Машина Паскаля

У 1673 р. німецький вчений Вільгельм Готфрід Лейбніц створив "арифметичний прилад" для додавання, множення і ділення 12-розрядних десяткових чисел.



Рис. 2. Машина Лейбніца

У Франції в 1799 р. Жозеф Марі Жакар винайшов ткацький верстат з перфокартами для задання узору – майже через два століття перфокарти стали масово використовуватися для введення інформації і програм в ЕОМ. У 1795 р. Гаспар Проні за дорученням французького уряду розробив технологічну схему обчислень (I етап – розробка числових методів, II етап – задання послідовності арифметичних дій (програмування), III етап – обчислення) для переходу на метричну систему мір, чим заклав основу для автоматизації обчислень.

У 1812 р. англійський математик Чарльз Беббідж запропонував ідею створення механічної різницевої машини для табулювання поліномів. Понсдавши ідею механічної машини з принципом програмного керування, згодом Чарльз Беббідж запропонував проєкт “аналітичної машини”, яка мала складатися з таких пристроїв:

- арифметичного (4 арифметичні дії, команда умовного переходу);
- запам’ятовуючого (1000 50-розрядних регістрів на зубчатих колесах);
- керування;
- вводу/виводу.

Але через велику складність цей проєкт так і не був реалізований.

У 1840 році англієць Томас Фуллер розробив і виготовив із перева обчислювач, який ґрунтувався на трійковій системі числення (рис. 3).

У 1875-1880 р. американець Герман Голеріт винайшов машину-табулятор, яка опрацьовувала інформацію з перфокарт.

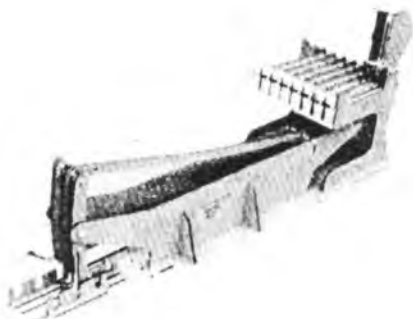


Рис. 3. Машина Фуллера, відтворена Марком Глоссером

Г. Голеріт заснував компанію, яка у 1911 р. стала називатися IBM (International Business Machines Corporation).

У 1906 р. Лі Ді Форест розробив вакуумний триод, а у 1918 р. професор М. А. Бонч-Бруєвич сконструював двостабільну електронну схему на двох вакуумних триодах – тригер.

У часи II світової війни в Німеччині, Англії і США велися роботи по створенню обчислювальних пристроїв для військових потреб. У Німеччині в 1937 р. Конрад Цузе розробив механічну машину Z1 (двійкова система числення, мантіса – 15 двійкових розрядів, порядок – 7 двійкових розрядів, оперативна пам'ять 64 слова, площа 2 кв.м). У 1938 р. Z2 – додано пристрій вводу даних і програми з перфорованої кінострічки, а арифметичний пристрій був побудований на телефонних реле. У цьому ж році К. Цузе отримав німецький патент Z23139/GMD Nr.005/021 на архітектуру обчислювача із процесором, оперативною пам'яттю програм і даних, побудованого на двійковій системі числення. У 1941 р. була створена машина Z3 (~ 2000 реле), а у 1945 р. – Z5. Для розрахунку крил ФАУ-1 і ФАУ-2 К.Цузе створив спеціалізовану релейну машину S2. Машина S3 використовувалася для керування ФАУ-1 і ФАУ-2 під час польоту. Роботи К.Цузе були засекречені, про них дізналися через кілька років після II світової війни. На рис. 4 подана фотографія машини Z3.

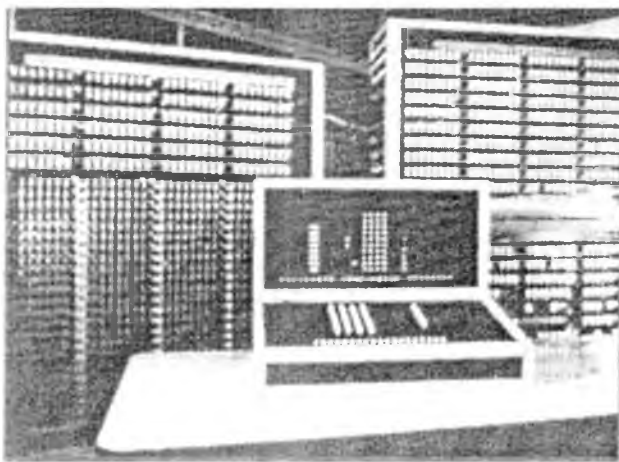


Рис. 4. Релейна ЕОМ Z3 (тактова частота 5.3 Гц, оперативна пам'ять 64 слова, 2 регістри, АЛПр. з плаваючою комою)

В Англії у 1936 р. Алан Тьюрінг запропонував теоретичні основи побудови обчислювальних машин (машина Тьюрінга), а у 1942-1943 рр. під керівництвом Алана Тьюрінга була створена спеціалізована електронна машина "Колосс" (2000 ламп) для розшифрування німецьких радіограм.

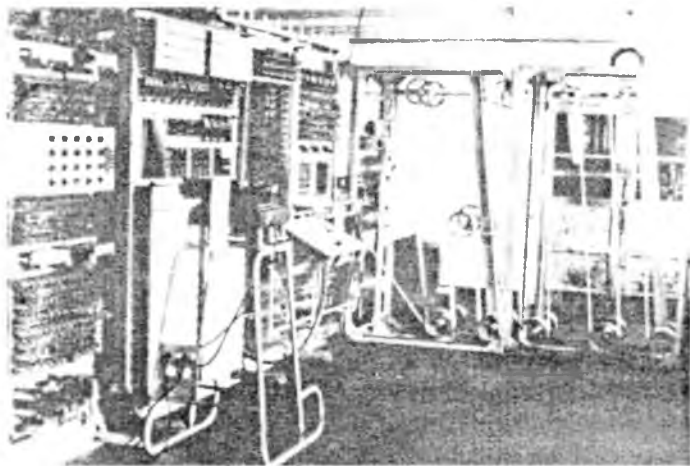


Рис. 5. Спеціалізована машина "Колосс"

У 1949 р. Моріс Уїлкс розробив електронну машину ЕДСАК послідовної дії з двійковою системою числення й оперативною пам'яттю на ртутних трубках. На основі ЕДСАК у 1953 р. була створена перша у світі комерційна серійна ЕОМ ЛЕО.



Рис. 6. Серійна ЕОМ "ЛЕО"

У США Дж. Мочлі і Дж. Преспер Еккерт у 1945 р. завершили розробку електронної машини ЕНІАК з десятковою системою числення (10 розрядів (десяткових), обсяг електронної пам'яті – 20 слів, площа 156 кв.м), яка була призначена для розрахунку таблиць стрільби артилерійських гармат (рух тіла під кутом до горизонту із врахуванням опору повітря, напрямку вітру, температури повітря тощо). У цьому ж році Дж. Мочлі, Дж. Преспер Еккерт і Джон фон Нейман розпочали розробку нової машини – ЕДВАК. У 1946 р. у науковій статті Неймана, Голдстайна і Бернса були викладені основні принципи побудови електронно-обчислювальних машин (така сама архітектура була викладена у патенті Z23139/GMD Nr.005/021 Конрада Цузе у 1937 р.):

- ЕОМ повинні ґрунтуватися на двійковій системі числення;
- програма повинна міститися в оперативному запам'ятовуючому пристрої, час доступу до ОЗП повинен бути співмірним із часом виконання арифметичних операцій;
- програма і числа повинні записуватись у двійковому коді;
- дані повинні розміщуватись в ОЗП;
- арифметичний пристрій будується на основі схем, які виконують операцію додавання (інші операції зводяться до багаторазового додавання), операції над усіма розрядами двійкового слова виконуються одночасно (паралельний принцип організації обчислювального процесу).

Слід зауважити, що ще у 1941р. Джон Атанасов і Кліффорд Беррі розробили проект електронної обчислювальної машини з двійковою системою числення і оперативною пам'яттю на конденсаторах (сучасна динамічна оперативна пам'ять є великим масивом мікроскопічних конденсаторів).

У 1947 р. Волтер Браттейн, Джон Бардін і Вільям Шоклі створили експериментальний зразок транзистора – основного елемента для ЕОМ II покоління.

У колишньому Радянському Союзі роботи з створення ЕОМ розпочав С.О.Лебедев – у грудні 1951р. у Києві (точніше, у Феофанії) держкомісія прийняла до експлуатації ЕОМ МЕСМ, яка складалася з пристроїв:

- арифметичного;
- управління;
- оперативної пам'яті;
- введення;
- виведення.



Рис. 7. Електронно-обчислювальна машина МЕСМ

ЕОМ МЕСМ була створена незалежно від розробок західних вчених, адже всі роботи по розробленню електронно-обчислювальних машин як у західних країнах, так і у СРСР були суворо засекречені.

Подальші роботи по створенню універсальних ЕОМ були перенесені з Києва в Москву. Наступний перелік ілюструє здобутки радянських вчених:

1956 р. – створена ЕОМ БЭСМ, яка має найвищу швидкість у Європі;

1958 р. – створена ЕОМ М40 з найвищою швидкістю у світі;

1959 р. – створена ЕОМ М50 з найвищою швидкістю у світі;

1967 р. – створена ЕОМ БЭСМ 6, яка в інтегральному виконанні мала швидкість один мільйон операцій в секунду.



Рис. 8. Електронно-обчислювальна машина БЭСМ 6



БЭСМ 6 випускалася 17 років. У 1972 р. її придбав Лондонський музей науки і техніки.

У 1958 році в Московському університеті була створена ЕОМ “Сетунь” (П. П. Брусенцов – апаратне забезпечення, Є.А. Жоголев – програмне забезпечення), яка ґрунтувалась на трійковій системі числення. Використання трійкової системи числення та особливої елементної бази (феритових кілець із обмотками – трансфлюксорів) дозволило створити надзвичайно просту архітектуру ЕОМ (і, як наслідок, надзвичайно надійну ЕОМ!). Промисловий випуск ЕОМ “Сетунь” було організовано на Казанському заводі лічильних машин, однак невисока відпускна ціна ЕОМ (встановлена директивно) не сприяла масовому випуску цих машин.

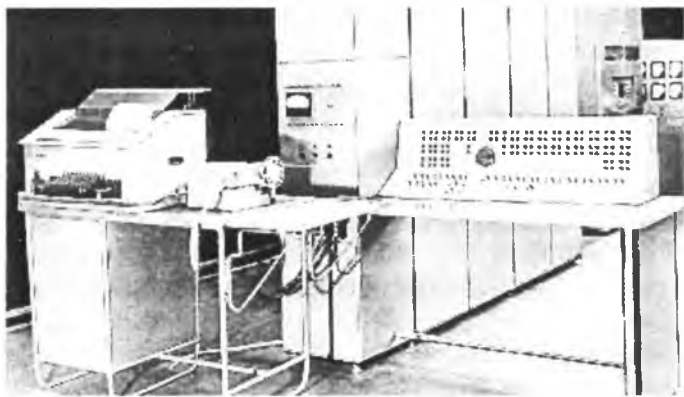


Рис. 9. Електронно-обчислювальна машина “Сетунь”

У 1970 р. у Московському університеті була створена ЕОМ “Сетунь-70” яка мала особливу систему команд (команди не мали фіксованої довжини. Команду можна було скласти з окремих частин – складів адресації і складів операції). У “Сетунь-70” було реалізовано принцип структурного програмування. Вона викликала величезний інтерес на Заході, але не була запущена у промислове виробництво у СРСР.

Після від’їзду С.О. Лебедева до Москви роботи по створенню ЕОМ у Києві продовжив В.М. Глушков, який очолив інститут кібернетики АН УРСР. Наступний короткий перелік (повний список розробок набагато більший) ілюструє розробки Київського інституту кібернетики:

1961 р. – створено напівпровідникову ЕОМ “Днепр” для керування технологічними процесами. Ця ЕОМ випускалася 10 років. Дві ЕОМ “Днепр” використовувались у Байконурі для керування великим екраном

під час польоту “Союз-Аполон” (основна частина розробок інституту кібернетики була скерована на створення керуючих ЕОМ різного, в основному спеціального призначення);

1963 р. – створено напівпровідникову ЕОМ “Промінь” для інженерних розрахунків (100 команд, російська мова програмування, вивід на електричну друкарську машинку);

1965 р. – створено напівпровідникову ЕОМ “Мир-1” для інженерних розрахунків (фірма IBM придбала одну “Мир-1” для того, щоб довести своїм конкурентам, що в СРСР був створений мікропрограмний принцип керування);

1969 р. – створено напівпровідникову ЕОМ “Мир-2” для інженерних розрахунків. На цій ЕОМ уперше у світі були виконані аналітичні обчислення.



Рис. 10. Електронно-обчислювальна машина “Мир-2”

Передові розробки радянських вчених були перервані після прийняття ЦК КПРС секретної постанови про заборону власних розробок універсальних ЕОМ і розробку аналогів лише двох ліній:

- IBM 360/370 – серія EC10xx;
- PDP 11 – серія CM.

Певний час розробки універсальних ЕОМ у всьому світі велися у трьох напрямках:

- супер-ЕОМ (Ельбрус, Cray);
- великі ЕОМ (IBM 360/370, EC10xx);
- міні-ЕОМ (PDP11, VAX, CM-x).

Після створення фірмою Intel у 1971 р. мікропроцесора I4004 (тактова частота 108кГц, 2300 транзисторів, 10 мікронна технологія) розпочалися розробки мікро-ЕОМ (LSI-11, microVAX, Електроніка-60, ДБК-2).

У 1975 р. на мікропроцесорі І8080 (6000 транзисторів, 10 мікронна технологія) був створений “електронний конструктор” Altair (ціна \$397), з якого кожний бажаючий міг зібрати мікро-ЕОМ з 4Кб оперативної пам’яті.

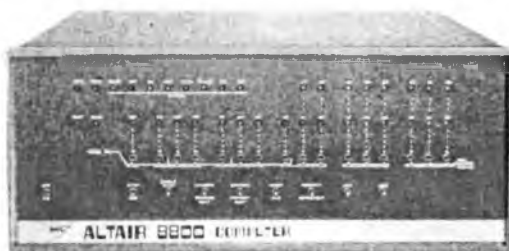


Рис. 11. Мікро-ЕОМ Altair

Перші зразки мікропроцесора І8080 продавалися за ціною 300 \$. У 1975 р. фірма Mos Technologies випустила мікропроцесор 6502, який продавався за ціною 25 \$. На цьому мікропроцесорі у 1976 р. Стефан Возняк і Стівен Джобс створили першу у світі мікро-ЕОМ Apple I, яка була орієнтована на **некваліфікованого користувача – персональний комп’ютер (personal computer)**. У 1977 р. розпочався промисловий випуск Apple II (процесор Motorola 6502, оперативна пам’ять 4Кб, пристрій для читання/запису інформації на ГМД, кольорова графіка, електронна таблиця VisiCalc).



Рис. 12. Персональний комп’ютер Apple II

Фірма Apple встановила обмеження на ціну своїх ПЕОМ – не більше \$6000 (модель Lisa, яка мала собівартість \$8000, комерційного успіху не мала). Поява на ринку Apple II привела до небаченого в історії США явища – службовці купували Apple за власний кошт для використання їх на роботі! У 1977 р. фірма Commodor International випустила ПЕОМ PET, а фірма Tandy Radio Shack – TRS-80 на мікропроцесорі Z-80. У 1982 р. фірма

Commodor Business Machines випустила ПЕОМ Commodor-65. У дослідницькому центрі Palo Alto Research Center корпорації Xerox у 1973 р. була створена ЕОМ Alto, а у 1982 р. – Star з “мишкою” і графічним інтерфейсом.

У 1981 р. році дванадцять інженерів фірми ІВМ під керівництвом Дона Естріджа виставили на загальний огляд свою розробку – ІВМ РС (процесор І8088 – 16-бітний з 8-бітною шиною даних і тактовою частотою 4,77 МГц, 16Кб (з можливістю розширення до 256Кб) оперативної пам'яті, вбудований Бейсік, один/два пристрої для роботи з 160Кб гнучкими магнітними дисками, монохромний або кольоровий дисплей Zenith, можливість використання кольорового телевізора замість дисплея). Ціна ІВМ РС з монохромним дисплеєм складала \$3000, з кольоровим – \$6000.



Рис. 13. Зовнішній вигляд ІВМ РС (фірмове позначення ІВМ 5051)

Архітектура ІВМ РС була оголошена відкритою – фірма ІВМ сподівалася у такий спосіб заощадити кошти на розробці периферійних пристроїв до своєї ПЕОМ – незалежні виробники, маючи документацію про архітектуру ІВМ РС, могли незалежно від фірми ІВМ (і за власні кошти) розробляти до ІВМ РС пристрої друку, дисплеї, пристрої читання/запису інформації на магнітні диски. Однак, попри очікуваний ефект, фірма ІВМ отримала несподіваний результат – незалежні виробники, користуючись відкритою архітектурою ІВМ РС, почали виробляти ІВМ-подібні ПЕОМ. Ці певні зміни в архітектурі ІВМ-подібної ПЕОМ, які робилися для того, щоб уникнути судових переслідувань з боку фірми ІВМ, узгоджувалися з BIOS (базовою системою вводу-виводу), і для користувача ПЕОМ ставала дуже подібною до ІВМ РС. Скромні можливості ІВМ РС (а отже і низька конкурентноздатність) змусили фірму ІВМ незабаром виставити ІВМ РС-ХТ (eXtended arhiTecture) та ІВМ РС-АТ (Advanced arhiTecture). Однак наздогнати конкурентів, які “осіддали” відкритість архітектури ІВМ РС, фірма ІВМ вже не могла. Не отримавши очікуваного результату

від відкритості архітектури IBM PC, фірма IBM спробувала закрити архітектуру своєї нової лінії IBM PS/2 (до речі, всі ПЕОМ фірми Apple мають закриту архітектуру), однак очікуваного результату це також не дало – світовий ринок вже був наповнений дешевшими IBM-подібними ПЕОМ зі співмірними або й кращими характеристиками. Скористався з відкритості архітектури IBM PC і Радянський Союз – у другій половині вісімдесятих років були розроблені ПЕОМ ЕС-1840 (Мінськ), Іскра-1030 (Смоленськ), Нейрон (Київ). Багатоплатна конструкція цих ПЕОМ стала причиною їх низької надійності. Особливе місце серед цих розробок займають “Поиск-1”, “Поиск-2” (копія IBM PC-XT), які вироблялися на Київському виробничому об’єднанні “Електронмаш”, і “Практик”, “ЕС 7978” (копія IBM PC-XT), які вироблялися на Канівському електромеханічному заводі “Магніт”. Слід наголосити на високій надійності ПЕОМ “ЕС 7978” і “Практик” – в одній із шкіл “ЕС 7978” пропрацювала понад 10 років без жодного ремонту (у радянські часи Канівський заводі “Магніт” виробляв прилади нічного бачення для танків і мав добре відпрацьовану технологію). Причиною низької надійності ПЕОМ “Поиск-2” у першу чергу було використання неякісних мікросхем оперативної пам’яті (білоруського виробництва). В/о “Електронмаш” і завод “Магніт” освоїли випуск ПЕОМ з процесором І80386, однак не змогли витримати конкуренції з невеличкими комп’ютерними фірмами, які почали масово збирати IBM-подібні ПЕОМ з комплектуючих, вироблених за кордоном (у першу чергу в південно-східній Азії).

У 1980 р. на ринку професійних мікро-ЕОМ з’явилася робоча станція Apollo (32-бітний мікропроцесор, власна Unix-подібна операційна система Domain OS з графічним інтерфейсом), яка була набагато дешевшою від великих і міні-ЕОМ (кілька десятків тисяч \$ замість сотень тисяч) і мала співмірну з ними обчислювальну потужність. Нижче подані результати виконання Форгран-тесту на великій ЕОМ, суперміні-ЕОМ і робочій станції:

- IBM-360 – 780 с;
- VAX 11/780 – 667 с;
- робоча станція з мікропроцесором Motorola 68020 (f-20МГц) – 748 с.

Поява у 1985 р. 32-бітного мікропроцесора І80386 уможливила створення співмірних за потужністю з робочими станціями, але значно дешевших IBM-подібних ПЕОМ.

Крім IBM-подібних ПЕОМ та ПЕОМ фірми Apple випускалися ПЕОМ інших ліній. Слід згадати Commodor, Atari, ZX-Spectrum, ПК-Львів,

Правець-8, БК-0010, Корвет, Агат. Ці ПЕОМ, як правило, випускалися в одному корпусі з клавіатурою, використовували 8-бітні мікропроцесори різних виробників і були непристосовані до модернізації, саме тому з появою більш потужних мікропроцесорів ці ПЕОМ просто зійшли з арени.

Для ілюстрації розвитку мікропроцесорів (і ПЕОМ) у таблиці подані основні характеристики мікропроцесорів, які використовувалися для побудови ПЕОМ.

Мікропроцесор	Розрядність	Швидкодія (тактова частота) МГц	Назва ПЕОМ
Intel 8080	8 біт	2	Altair, ПК-Львів
Z80A	8 біт	6-8	ZX-Spectrum, Robotron, Агат
Intel 8088	8/16 біт	4.77	IBM-PC, IBM-PC/XT
K1810BM 88 (аналог I8088)	8/16 біт	4.77	Поиск-1
K1810BM 86 (аналог I8086)	16 біт	8	Поиск-2, Практик, ЕС-7978
Intel 80286	16 біт	8-12	IBM-PC/AT
Motorola 68000	16 біт	6-16	Apple, Macintosh
Intel 80386	32 біт	16-40	IBM-PC.386
Motorola 68020	32 біт	16-20	Macintosh***** Беста (виробн. СРСР)
Intel 80486	32 біт	16-120	IBM-PC.486
Motorola 68030	32 біт	16-50	Macintosh*****
Pentium I	32/64 біт	60-133	IBM-PC.PI
Motorola 68040	32/64 біт	16-60	Macintosh*****
Pentium II	32/64 біт	233-500	IBM-PC.PII
Power PC	32/64 біт	100-133	Power Macintosh
Pentium III	32/64 біт	450-1000	IBM-PC.PIII
Power PC G3	32/64 біт	350-500	iMac
Power PC G4	32/64 біт	400-733	Power Macintosh
Pentium IV	32/64 біт	3800	IBM-PC.PIV
Power PC G5	32/64 біт	До 2700	Power Macintosh, iMac, eBook, Power Book

Примітка: \*\*\*\*\* означає фірмове позначення моделі

Сучасні ПЕОМ можна розділити на дві групи:

- ПЕОМ фірми Apple (Macintosh);
- IBM-подібні ПЕОМ від різних виробників (у тому числі українських).

ПЕОМ фірми Apple завжди відрізнялися високими споживчими характеристиками, які забезпечувалися високим рівнем схемних рішень та високою якістю програмного забезпечення. Авторам доводилося спостерігати у роботі Mac OS 5 з графічним інтерфейсом, яка завантажувалась з 3,5" дискети (порівняйте 700 Кб з мінімумом у 100 Мб для Windows-95). Політика закритої архітектури (Apple та Macintosh випускають лише 4 заводи в світі) дозволяла випускати ПЕОМ лише високої якості. Розробники програмного забезпечення, які бажають писати програми для Apple, повинні пройти обов'язкову сертифікацію (у такий спосіб забезпечується належний рівень програмного забезпечення для Apple). Однак за високу якість доводиться розплачуватися – ціни на ПЕОМ від Apple у 2-3 рази вищі від цін співмірних за потужністю IBM-подібних ПЕОМ. В Україні ПЕОМ Apple використовують, як правило, у видавництвах (якщо видавництво спроможне придбати таку дорогу техніку).

IBM-подібні ПЕОМ сьогодні в Україні займають понад 90% парку персональних комп'ютерів. Це зумовлено більш низькою ціною і відкритістю архітектури.

## Розділ І. ТЕОРЕТИЧНІ ОСНОВИ ФУНКЦІОНУВАННЯ ЕОМ

Почнемо з визначення деяких понять.

● ЕОМ – це, мабуть, найпоширеніше поняття до зовсім недавнього часу. Спочатку воно стосувалося перших електронних автоматичних пристроїв для опрацювання інформації. Однак з часом на ЕОМ почали покладати функції інформаційного плану: зберігання, пошук, сортування, опрацювання інформації (в тому числі графічної, текстової, образної (передавання зображень), звукової). Розрізняють спеціалізовані ЕОМ; комплекси ЕОМ; універсальні ЕОМ; міні-, мікро-ЕОМ.

● ОС – обчислювальна система – пристрій, що сприймає інформацію у вигляді даних, які можуть бути зображені в аналоговій або цифровій формі, зберігає дані, опрацьовує їх з великою швидкістю, пересилає дані всередині системи через лінію зв'язку і видає результат цих дій як інформацію. (Не треба плутати з поняттям операційна система, що також має скорочене позначення ОС). Обчислювальні системи можуть бути:

- аналоговими (безперервної дії);
- цифровими (дискретними);
- гібридними.

● Комп'ютер (англ. *computer* – обчислювач) – спочатку в англійських країнах, а тепер майже всюди, назва електронної обчислювальної машини. Головно слугує для передавання, опрацювання і зберігання інформації. Частіше вживають термін “персональний комп'ютер”, або скорочено ПК (на думку авторів, доречніше вживати термін ПІОМ).

### 1.1. Поняття про інформацію

Про сучасні комп'ютери, їхню будову, можливості та принципи функціонування йтиметься дещо пізніше, спочатку ж розглянемо, що таке інформація і як її можна відображати в ЕОМ. Слово *інформація* походить від лат. *informatio*, що означає роз'яснення, виклад, обізнаність. Це одне з найзагальніших понять науки. Поняття інформації – одне з



головних понять кібернетики (подібно до поняття енергії у фізиці). Розділ науки, присвячений технічним проблемам інформації (окрім семантичних та прагматичних), називають *теорією інформації*. Різні способи відображення інформації, спеціально пристосовані для конкретних випадків, пов'язаних із передаванням, зберіганням та опрацюванням інформації, розглядає теорія кодування.

В ЕОМ маємо справу з дискретними повідомленнями, або машинними одиницями інформації. Насамперед постає питання, як виразити, виміряти і передати інформацію? Уперше цю задачу розв'язано у працях Клода Шеннона [29], де запроваджено поняття кількості інформації. Дещо спрощено схема викладу цього поняття може бути такою.

Нехай нам потрібно скласти слово як комбінацію з букв алфавіту. Алфавіт містить  $P$  букв (чи символів), а слово –  $x$  символів. Очевидно, таких слів можна скласти  $N$  (правда, невідомо, чи всі вони будуть мати мовний зміст):

$$N = P^x \text{ (кількість розміщень)}. \quad (1.1)$$

Припустимо, що кількість інформації у якомусь повідомленні пропорційна до його довжини. Тоді з (1.1) після логарифмування отримаємо

$$\log_a N = x \log_a P \quad (1.2)$$

Величину  $x \log_a P$  визначимо як кількість інформації  $I$ . Знайдемо найменше  $I$ , яке можна було б прийняти за одиницю інформації. Очевидно, це буде при  $P=2$ , бо при  $P=1$ ,  $I=0$ . Отже,

$$I_{\min} = x \log_2 P. \quad (1.3)$$

Найпростіше слово складається з однієї букви, тоді

$$I = \log_2 P. \quad (1.4)$$

Це і є кількість інформації найпростішого слова.

Вибір основи логарифма – довільний. Зважаючи на те, що в обчислювальних системах головно послугуються двійковою арифметикою, прийемо основу логарифма  $a=2$ . Тоді

$$I = \log_2 P. \quad (1.5)$$

Ця одиниця інформації у двійковому алфавіті називається *біт* і може мати значення 0 і 1 (походить від англ. *binary digit* і, як стверджує К. Шеннон [29], це слово запропонував уживати відомий американський вчений-статистик Джон Тьюкі). Отже, кількість інформації у будь-якому слові двійкового алфавіту завжди дорівнює кількості бітів у ньому, тобто

$$I = x \log_2 P = x, \quad (1.6)$$

а кількість можливих повідомлень  $N$  легко визначити з (1.2):

$$\log_2 N = x \log_2 2, \text{ тобто } N = 2^x. \quad (1.7).$$

Ще раз зазначимо, що таке поняття кількості інформації виникло з теорії зв'язку і, по суті, застосовне саме до них.

*Приклад:* Кількість можливих слів, які містять три біти, є 8. Ось вони: 000, 001, 010, 011, 100, 101, 110, 111. Це не що інше, як числа шістнадцяткової системи числення.

Кількість можливих слів, які містять чотири біти  $N = 2^4 = 16$ . Це такі комбінації: 0000, 0001, 0010, 0011, 0100, 0101, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111, тобто числа шістнадцяткової системи числення.

Біт є найменшою одиницею вимірювання інформації. На практиці частіше використовують похідні одиниці: *байт*, *кілобайт* (*Кбайт*), *мегабайт* (*Мбайт*), *гігабайт* (*Гбайт*), *терабайт* (*Тбайт*), *петабайт* (*Пбайт*), *ексабайт* (*Ебайт*). Між ними існують такі співвідношення:

- 1 байт =  $2^3$  бітів = 8 бітів (може бути  $2^8 = 256$  різних байтів);
- 1 Кбайт =  $2^{13}$  бітів = 1024 байти (скорочено позначають К);
- 1 Мбайт =  $2^{23}$  бітів = 1024 Кбайти;
- 1 Гбайт =  $2^{33}$  бітів = 1024 Мбайти;
- 1 Тбайт =  $2^{43}$  бітів = 1024 Гбайти;
- 1 Пбайт =  $2^{53}$  бітів = 1024 Тбайти;
- 1 Ебайт =  $2^{63}$  бітів = 1024 Пбайти.

До машинних одиниць інформації належать також: слово, запис, блок, файл. Деякі машинні одиниці мають аналоги з натуральними одиницями інформації: розряд, символ, поле, масив, запис.

## 1.2. Основи алгебри логіки

Усі пристрої ЕОМ складаються з елементарних логічних схем. Їхнє функціонування ґрунтується на законах і правилах алгебри логіки, яка оперує двома поняттями: істина і фальш. На честь її винахідника, англійського математика Джорджа Буля, алгебру логіки ще називають булевою алгеброю. Основу цієї алгебри становлять дві бінарні операції (*кон'юнкція* та *диз'юнкція*) і одна унарна (*заперечення*). Крім цих трьох, вводять і інші, однак доведено, що будь-яку з них можна виразити за допомогою формули, у якій використано тільки три базові. (Наприклад, для функції від двох змінних таких логічних функцій є 16).

Мовою алгебри логіки будь-яку функцію зображають у вигляді таблиці відповідності всіх можливих логічних змінних та вихідних логічних функцій. Це так звана *таблиця істинності*.

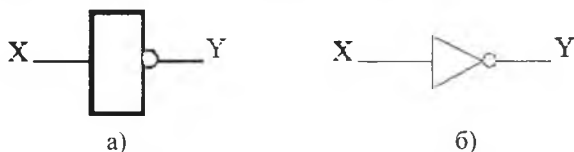
Розглянемо головні логічні функції.

**Логічне заперечення.** Позначають НЕ.

**Означення.** Логічне заперечення НЕ змінної  $X$  є логічна функція  $Y$ , яка істинна тільки тоді, коли  $X$  хибна, і навпаки. Інша назва: *інверсія*.

У символах алгебри логіки записують:  $Y = \bar{X}$ .

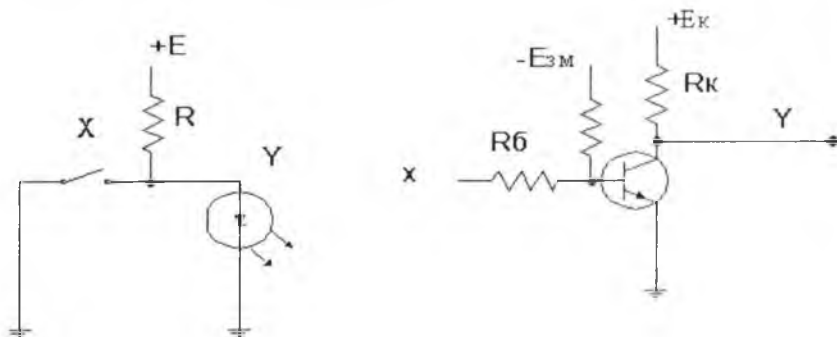
Графічно на схемах позначають кружечком на вході, чи виході логічного символу (а – у системі ЄСКД, б – в американському стандарті):



Таблиця істинності:

X	0	1
Y	1	0

Схемна реалізація – інвертор.

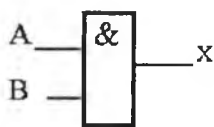


**Логічне множення.** Позначають І.

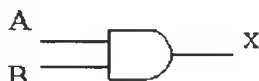
**Означення.** Логічним множенням двох змінних  $A$  і  $B$  є логічна функція  $X$ , яка істинна тільки тоді, коли одночасно істинні входні змінні. Інша назва: *кон'юнкція*.

Запис:  $X = A \times B$ , або  $X = A \wedge B$ .

Графічне зображення (а – у системі ЄСКД, б – в американському стандарті):



а)

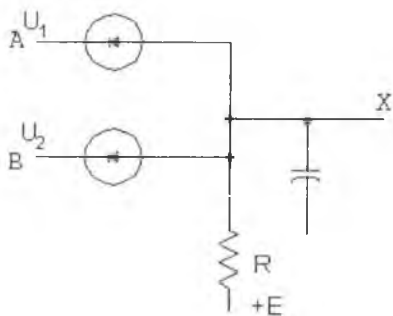
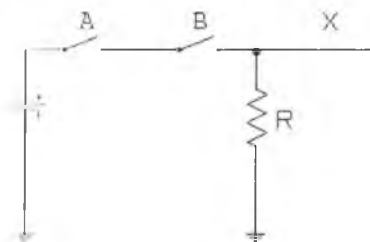


б)

Таблиця істинності:

A	0	0	1	1
B	0	1	0	1
X	0	0	0	1

Схемна реалізація:



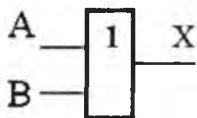
**Логічне додавання.** Позначають АБО.

**Означення.** Логічною сумою змінних  $A$  і  $B$  є логічна функція  $X$ , яка істинна, якщо хоча б одна із вхідних величин істинна.

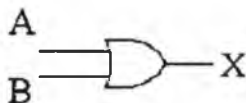
Інша назва: *диз'юнкція*.

Запис:  $X = A + B$ , або  $X = A \vee B$ .

Графічне зображення (а – у системі ЄСКД, б – в американському стандарті):



а)

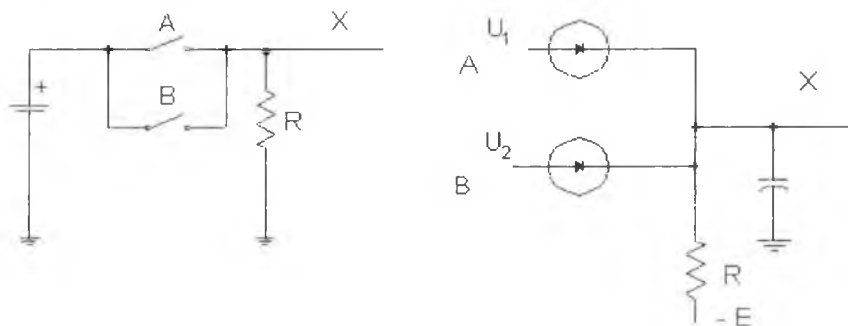


б)

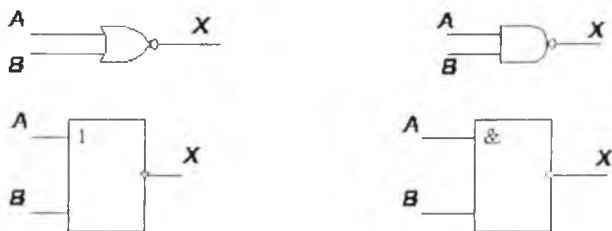
Таблиця істинності:

A	0	0	1	1
B	0	1	0	1
X	0	1	1	1

Схемна реалізація:



У реальних схемах символічні ключі замінюють діодами, транзисторами чи просто використовують напівлені  $p-n$  – переходи у схемах більшої інтеграції. Схеми також можуть об'єднуватись відповідно до потреби реалізації певної логічної функції. Наприклад, дуже часто застосовують елементи АБО-НЕ (елемент Пірса) чи І-НЕ (елемент Шеффера):



Функції, утворені з логічних змінних, можна перетворювати відповідно до правил або законів алгебри логіки.

Приклад.

1. Логічний добуток будь-якого аргументу на нуль завжди дорівнює нулю (використовують для “маскування” вхідних змінних).

2. Логічний добуток будь-якого аргументу на 1 дорівнює значенню самого аргументу.

3. Логічний добуток аргументу з його інверсією дорівнює нулю.

4. Правила де Моргана (закони інверсії):

$$A \times B = \overline{\overline{A} + \overline{B}};$$

$$\overline{A + B} = \overline{A} \times \overline{B}$$

тощо.

Ці правила і закони застосовують для спрощення логічних функцій і введення їх до вигляду, що полегшує схемну реалізацію. Для забезпечення мінімізації логічних рівнянь використовують запис вхідних даних у вигляді діаграм Карно (Karnaugh map).

На рисунку нижче показана діаграма Карно для трьох змінних.

	B		$\overline{B}$	
A	6	7	5	4
$\overline{A}$	2	3	1	0
	C			$\overline{C}$

Кожна клітинка діаграми відповідає логічному добутку прямого чи інверсного значення змінних, які присвоєно стовпцеві або рядкові, на перетині яких вона розміщена. Наприклад, клітинка з номером 7 є на перетині рядка зі значенням змінної  $A$  і стовпця зі значенням змінних  $B$  і  $C$  та відповідає логічному добутку  $AB\overline{C}$ , клітинка з номером 0 відповідатиме значенню  $\overline{A}\overline{B}\overline{C}$ . У діаграмах Карно значення змінних присвоюються так, щоб сусідні клітинки відрізнялися між собою тільки на одну змінну. Після цього згідно з описом логічної функції кількох змінних у клітинки записують значення логічних добутків з таблиці істинності. Якщо добутки є в сусідніх клітинках, то з загального виразу можна виключити одну змінну, яка трапляється у прямому та інверсному кодах. Якщо добутки утворюють квадрат, то з загального виразу можна виключити дві змінні.

Діаграми Карно ефективні, якщо число змінних не перевищує шести.

### 1.3. Елементна база ЕОМ

Для реалізації логічних схем та елементів пам'яті будь-якої ЕОМ використовують конкретні фізичні пристрої: електромагнітні реле, електронні лампи, транзистори, трансфлюксори, мікросхеми різного ступеню інтеграції. Кожний з цих пристроїв, слугуючи базою для побудови головних логічних елементів та вузлів, у свій час визначав цілі покоління ЕОМ. Наведемо головні характеристики та схемні позначення цих пристроїв.

**Електромагнітні реле:** базові елементи для побудови перших ЕОМ (у деяких джерелах відлік поколінь ЕОМ ведуть від електронних ламп). Конструктивно складаються з котушки на магнітопровідному осерді та контактної групи. При протіканні через котушку струму (відповідає логічній одиниці) осердя намагнічується і перемикає контактну групу. Відсутність струму в котушці (відповідає логічному нулеві) приводить контактну групу у вихідний стан. Час спрацювання реле значний – десятки мсек, споживана потужність велика – 0,5 – 5 Вт.

**Електронні лампи:** базові елементи для побудови ЕОМ першого покоління. Виготовляються у вигляді скляного або металевого балона з виведенням елементів керування на цоколі. Найпоширеніший тип електронної лампи у побудові ЕОМ – тріод, який складається з трьох елементів: катода, анода та сітки. Катод при розігріванні здатний продукувати електрони, які за сприятливих умов можуть досягати іншого електрода – анода, створюючи струм в електричному колі. Третій електрод – сітка, виконує функцію регулювання цього струму. Наприклад, наявність негативного потенціалу на сітці тріода (логічний нуль) забороняє протікання анодного струму, а наявність позитивного (логічна одиниця) дозволяє протікання струму. Така електронна ключова схема дозволяє легко реалізувати головні логічні елементи та елементи пам'яті. Частота спрацювання схем на електронних лампах досягає 10 МГц.

**Транзистори:** напівпровідникові базові елементи для побудови ЕОМ другого покоління. Транзистори виготовляють у металевих або пластмасових корпусах, переважно вони мають три виводи, які з'єднуються всередині корпуса з електродами. Струм, який протікає між двома з трьох електродів (емітер – колектор), регулюється відповідно до зміни струму чи напруги на третьому (базі). Транзистори однаково успішно використовують як для підсилення сигналів так і для побудови схем перемикання.

Частота спрацювання схем на транзисторах сягає десятків МГц, а споживана потужність – долі вата.

**Трансфлюксори:** логічні та запам'ятовувальні елементи з магнітного матеріалу з прямокутною петлею гістерезису, які функціонують за принципом перерозподілу магнітного потоку. Виготовляють методом пресування феритного порошку у вигляді кільця з двома нерівними отворами (див. [12]). Записування і зчитування інформації здійснюється за допомогою координатних шин, які приєднуються до електронних схем. Частота зчитування з трансфлюксора обмежена нагріванням магнітного матеріалу і, як правило, не перевищує 1 МГц, а частота записування приблизно у 2-3 рази менша. Використовувалися як елементи пам'яті операційних запам'ятовувальних пристроїв ЕОМ другого і третього покоління.

**Мікросхеми малого та середнього ступеня інтеграції:** елементна база третього та четвертого покоління ЕОМ. Переважно, це ГЛ – схеми (транзисторно-транзисторна логіка), головними комутуючими елементами яких є біполярні транзистори. В одному корпусі міститься від кількох одиниць до кількох десятків транзисторів з малим споживанням потужності та високою швидкодією.

**Мікросхеми великого ступеня інтеграції** передбачають розміщення в одному корпусі 500 і більше транзисторів, які виготовлені за біполярною технологією, і 1000 і більше елементів, виготовлених за МДП-технологією (метал-діелектрик-напівпровідник). Надвеликі інтегральні схеми містять понад 10 000 елементів. Сучасні надвеликі процесорні інтегральні схеми містять сотні мільйонів транзисторів.

### Головні логічні елементи та вузли

Елементами ЕОМ називають пристрої, які виконують логічні функції, запам'ятовують інформацію, перетворюють її, а також формують та підсилюють сигнали.

З деякими найпростішими елементами ми вже ознайомлені. Це інвертор, пристрій для логічного додавання і множення. Відомо також, що використовуючи їхню різноманітну комбінацію, можна будувати складні логічні схеми.

Головним елементом статичної пам'яті ЕОМ є *тригери* – логічні пристрої, які мають два стійких стани. Для перемикавання тригерів з одного стану в інший використовують входні логічні схеми. За способом перемикавання розрізняють такі тригери:



- *RS*-тригери з роздільним установленням 0 і 1;
- *D*-тригер із затримкою;
- *JK*-тригери (універсальні);
- *T*-тригери (лічильні).

Ці назви походять від перших букв вхідних сигналів: *S* (set – встановити); *R* (reset – вимкнути); *T* (toggle – релаксатор); *J* (jerk – різко увімкнути); *K* (kill – різко вимкнути); *D* (delay – затримка). Вихідний сигнал тригера прийнято позначати буквою *Q*.

### ***RS*-тригери з роздільним установленням 0 та 1**

Найпростіший запам'ятовувальний елемент має два входи: *S* і *R*. У разі комбінації сигналів  $S=1, R=0$  тригер буде встановлений в одиничний стан, тобто  $Q=1$ .

Якщо ж комбінація вхідних сигналів  $S=0, R=1$ , то тригер встановиться в нуль,  $Q=0$ . Комбінація  $S=0, R=0$  залишить тригер у попередньому стані, а от комбінація  $S=1, R=1$  буде невизначеною, отже, забороненою для такого елемента ( $Q = \bar{Q} = 0$ ).

Функціонування *RS*-тригера описується виразом

$$Q(t+1) = S(t) \vee Q(t) \bar{R}(t),$$

де  $S(t)R(t)=0$ ,  $t$  – момент часу, який передуює зміні стану.

Такі тригери реалізують на вже відомих нам логічних елементах І-НЕ, АБО-НЕ, з'єднуючи їхні входи і виходи навхрест. На схемах їх зображають прямокутником, у якому з лівого боку позначають вхідні сигнали *S*, *R*, а з правого – вихідні. Вхід *S* називають установлювальним, а вхід *R* – скидальним. Приклади таких *RS*-тригерів показано на рис. 1.1.

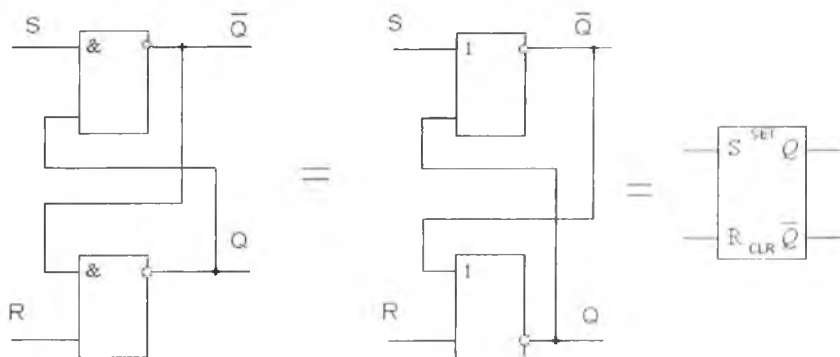


Рис. 1.1. *RS*-тригери на базі елементів І та АБО

Іноді вводять синхронізацію зміни стану тригера (позначають вхід  $C$  (clocking – синхронізація)). При  $C=1$  тригер перемикається за законом  $RS$ -тригера, при  $C=0$  він зберігає попередній стан.

### ***D*-тригер (затримки)**

Такий тригер має один інформаційний вхід і вхід для синхроімпульсу. Головне призначення такого тригера – затримка і збереження сигналу, який подають на вхід за умови  $C=1$ . На рис. 1.2 показана схема синхронізованого двотактного  $D$ -тригера, який затримує сигнал на один період; і його описує формула:

$$Q(t+1) = D(t).$$

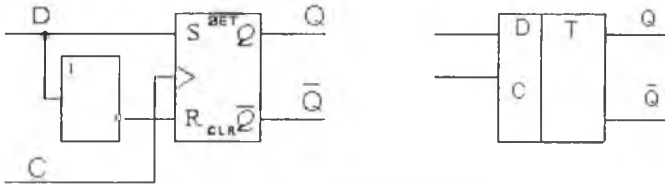


Рис. 1.2. Синхронізований  $D$ -тригер

Різновидом  $D$ -тригера є  $DV$ -тригер ( $V$ -value – вентиль), у якому через вхід керування  $V$  дозволено перемикання, або тригер не реагує на перемикання при  $V=0$ .

### ***JK*-тригер (універсальний)**

Працює за принципом  $RS$ -тригера, проте в ньому комбінація  $J=K=1$  не заборонена, у цьому разі він змінює свій стан на протилежний. Переважно його реалізують за двоступеневою схемою (рис. 1.3).

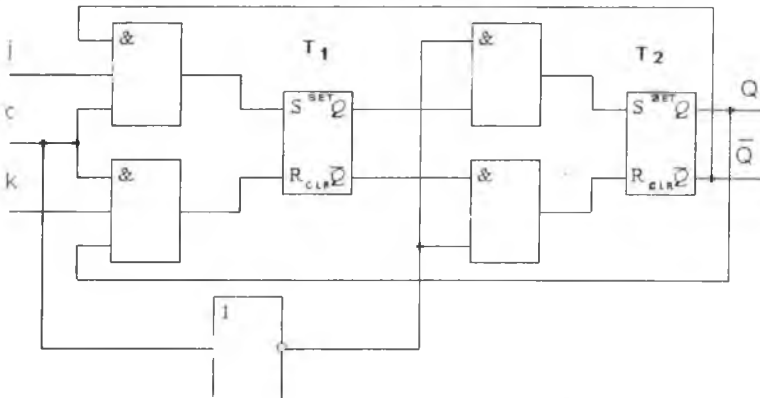
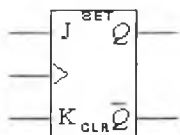


Рис. 1.3. Схемна реалізація  $JK$ -тригера

На схемах позначають так:



При  $C=1$  відбувається записування інформації в  $T1$ , при  $C=0$  – переписування інформації з  $T1$  в  $T2$ . Функція переходів  $JK$ -тригера за умови  $J=K=1$  має вигляд

$$Q(t+1) = Q(t)\bar{K}(t) \vee \bar{Q}(t)J(t).$$

### **T-тригер (лічильний)**

Цей тригер змінює свій стан у разі надходження кожного входного імпульсу. Він може бути реалізований на базі  $JK$ -тригера. При значеннях змінних  $J=K=1$  сигналом  $C$  можна змінювати стан тригера. Позначають як на рис. 1.4.

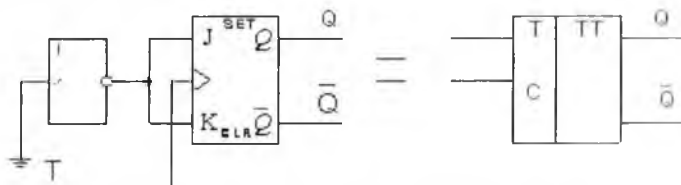


Рис. 1.4. Схемна реалізація лічильного тригера

T-тригер реалізує таку функцію:

$$Q(t+1) = Q(t)\bar{T}(t) \vee \bar{Q}(t)T(t).$$

За допомогою описаних логічних пристроїв можна будувати різні блоки чи пристрої ЕОМ. До них належать: регістри, лічильники, шифратори, дешифратори, мультиплексори, суматори та арифметико-логічні пристрої як цілочислові так і плаваючої арифметики.

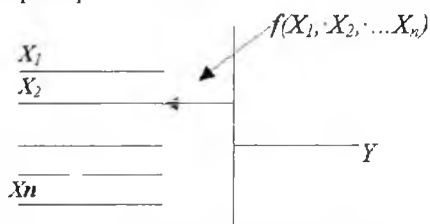


Рис. 1.5. Схема комутування входних сигналів  $X$  на вихідну лінію  $Y$  за допомогою функції  $f(X_1, X_2, \dots, X_n)$

**Селектор/Мультиплексор** – пристрій, який виконує передавання даних від однієї з входних ліній у вихідну (рис. 1.5). Вхідну лінію вибирають кодом, котрий вказує, який із входів потрібно приєднати на вихід. Цей процес називається мультиплексуванням у часі.

**Регістр** – це вузол ЕОМ, який виконує тимчасове збереження та перетворення інформації. Регістри будують на основі тригерних схем. Кількість тригерів визначає розрядність слів, які записують чи зберігають у регістрі. Регістри є послідовні і паралельні, одно- і двотактні, зсувні і перетворювальні. Розглянемо кілька прикладів схем регістрів.

Побудуємо регістр, наприклад, трирозрядний. Очевидно, що для цього треба мінімум три тригери, які будемо використовувати для запису кожного розряду.

Використаємо звичайні *RS*-тригери. Трохи добудуємо схему ліворуч і праворуч. Ліворуч організуємо синхронний запис інформації, а праворуч – формування прямого та оберненого коду. Перед записом інформації всі тригери сигналом керування *R* установимо в нуль  $\Rightarrow (Q=0)$ . Запис у тригери виконується за тактовим імпульсом  $Ti_1$ . Власне ці два сигнали (*R* і  $Ti_1$ ) визначають тип регістра, який називають двотактовим паралельної дії (рис. 1.6).

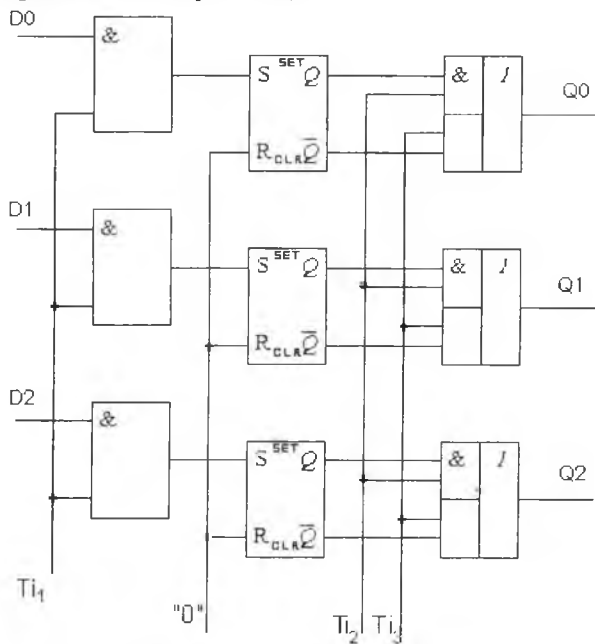


Рис. 1.6. Двотактовий регістр паралельної дії

Код регістра видають за допомогою другого і третього тактових імпульсів:  $Ti_2$  – прямий код,  $Ti_3$  – обернений код.

Якщо на вхід можна подавати парафазний код (тобто вхідне значення подають у прямому й оберненому коді), то відпадає потреба у такті установлення в "0", тобто отримаємо однотактний паралельний регістр.

Регістр є дуже зручним пристроєм для "зсування" інформації праворуч чи ліворуч або перетворення послідовного коду в паралельний (рис. 1.7). Для цього тригери регістра потрібно з'єднати послідовно через невелику затримку.

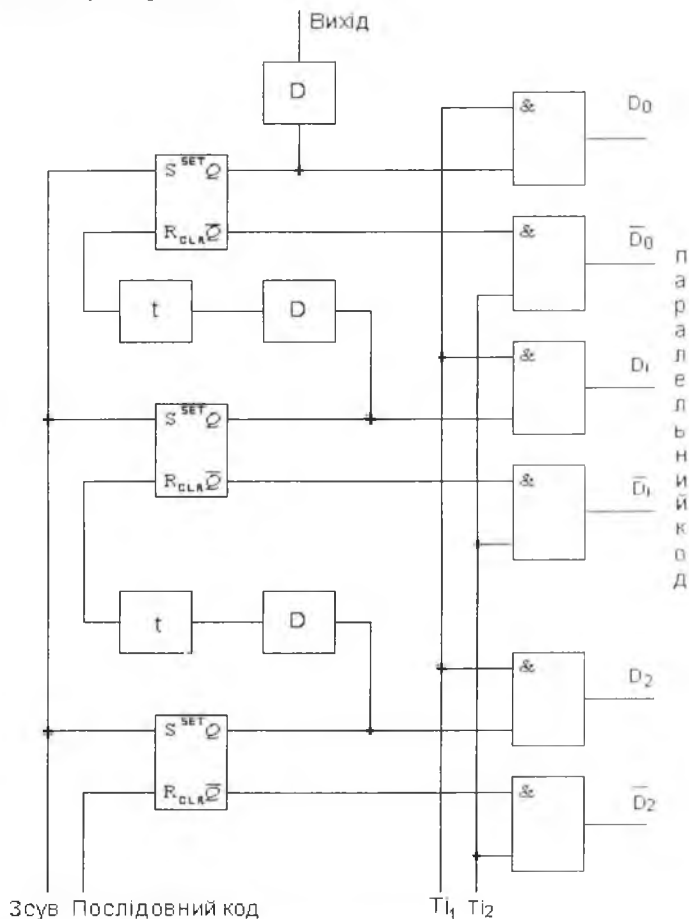


Рис. 1.7. Регістр для перетворення послідовного коду в паралельний.

$D$  – диференціальна ланка, яка формує імпульс для встановлення тригера по входу  $R$ ;  $t$  – невелика лінія затримки

На вхід  $R$  першого тригера надходить послідовний код, наприклад, 1011. Входи  $S$  усіх тригерів об'єднані і на них надходить імпульс зсуву, який одночасно виконує функцію встановлення в "0" регістра. Тригери спрацьовують тільки від додатних імпульсів. Це означає, що встановлення наступного тригера можливе лише в разі перекидання попереднього з "1" у "0". Незаважко помітити, що через три кроки послідовний код 1011 буде занесений у регістр і може бути прочитаний у паралельному (прямому або інверсному) коді.

Цієї розглянемо роботу однотактового регістра зі зсуванням числа праворуч (рис. 1.8).

Тут для спрощення схеми ланки зчитування числа пропущено. Для зсуву числа потрібно подати  $n$  імпульсів зсуву.

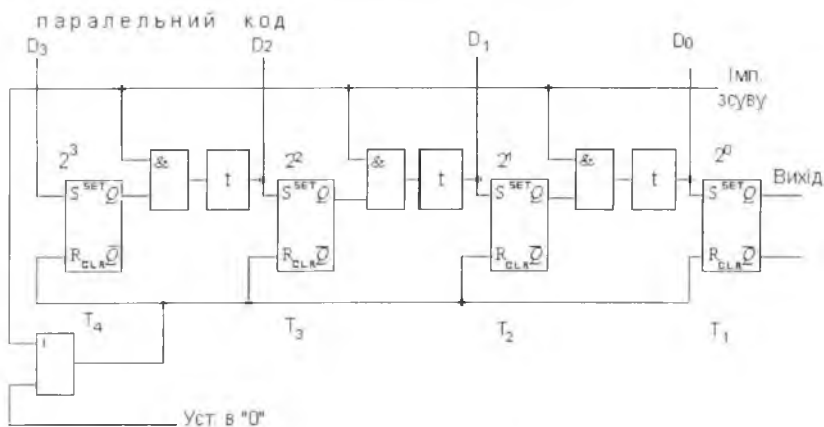


Рис. 1.8. Однотактовий регістр для зсування коду праворуч

Нехай у регістрі є код 1011. Стан тригерів після кожного імпульсу зсуву  $t$  буде змінюватися відповідно до таблиці:

Імпульс	$t_1$	$t_2$	$t_3$	$t_4$	$t_2 = t_1 + t_{затр}$ $t_3 = t_2 + t_{затр}$ $t_4 = t_3 + t_{затр}$
$T_4$	1	0	0	0	
$T_3$	0	1	0	0	
$T_2$	1	0	1	0	
$T_1$	1	1	0	1	

Якщо використовувати парафазний (такий, у якому присутні одночасно прямі та інверсні значення) код, то потреби у лініях затримки немає.

Затримку можна виконати також за допомогою тригера, що значно підвищує надійність роботи схеми. Зсув можна робити як праворуч, так і ліворуч. Такі регістри називають *реверсивними*.

Якщо вихід  $T_i$  подати на вхід  $T_{i+1}$ , то легко помітити, що в разі подання імпульсів зсуву інформація в регістрі буде циркулювати. Такі регістри називають *кільцевими*.

**Лічильник** – пристрій, призначений для підрахунку кількості імпульсів. Лічильники бувають підсумовувальні, віднімальні та реверсивні. Реверсивний лічильник залежно від перекомутації може бути підсумовувальним або віднімальним. Будують лічильники на основі тригерів, використовують їх у пристроях керування та в арифметичних пристроях для підрахунку номерів команд, кількості циклів програми, кількості тактів у разі множення і ділення, а також як суматори. Приклад 4-розрядного підсумовувального двійкового лічильника показано на рис. 1.9.

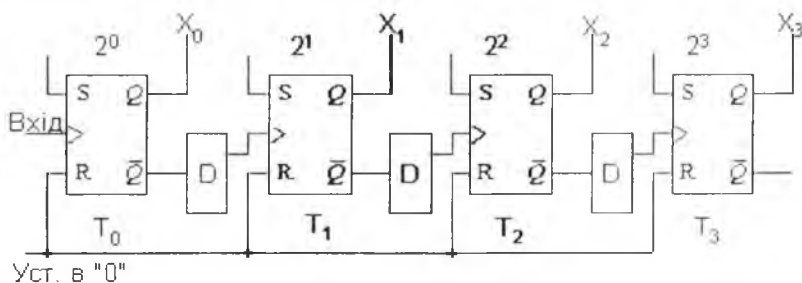


Рис. 1.9. Схема 4-розрядного підсумовувального двійкового лічильника

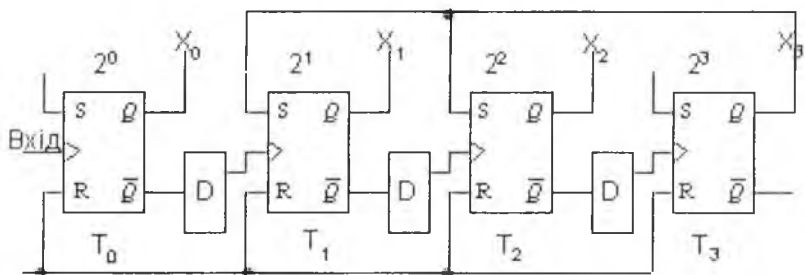
Такий лічильник порахує до 16 (1111) і знову стане в "0". Максимальна кількість імпульсів, яку може підрахувати двійковий лічильник, що складається з  $n$  розрядів, дорівнює  $2^n - 1$ .

Віднімальний лічильник отримаємо тоді, коли сигнал зніматимемо з інверсного виходу тригера.

Лічильники, які однаково можна використовувати як для додавання, так і для віднімання імпульсів, називають *реверсивними*.

Входи  $S, R$  застосовують для занесення початкового коду в лічильник або скидання в "0".

Іноді потрібно отримати значення  $K \neq 2^n$ . Тоді будують спеціальні перерахункові схеми, де від певних розрядів уводять обернений зв'язок. Перерахункові схеми з  $K=10$  називають декадними лічильниками, які застосовують для побудови десяткових лічильників (рис. 1.10).



Уст. в "0"

Рис. 1.10. Декадний лічильник

У разі подання перших семи імпульсів схема працює у звичайному режимі. Восьмий імпульс установить тригери  $T_0$ ,  $T_1$  і  $T_2$  в стан "0", а тригер  $T_3$  – у стан "1". У цьому випадку внаслідок оберненого зв'язку сигнал перенесення з  $T_3$  встановить в одиничний стан  $T_1$  і  $T_2$ : отримаємо стан лічильника 1110. Дев'ятий імпульс установить в "1"  $T_0$ , а десятий скине в "0" усі тригери схеми. Після цього схема знову готова рахувати до 10.

**Шифратор** – це вузол ЕОМ, що виконує операцію формування відповідного двійкового коду в разі появи сигналу на одному із входів шифратора. В обчислювальній техніці застосовують переважно багатоімпульсні шифратори, які дозволяють кодувати дані при записуванні програм на носії інформації чи дані, виражені якоюсь фізичною величиною (напр., напругою).

**Дешифратор** – пристрій, який виконує перетворення  $n$ -розрядного двійкового коду в однорозрядний з основою  $p=2^n$ , тобто функцію, обернену до шифрування. Використовують його у багатьох пристроях, зокрема у пристроях керування, для розшифровування коду операції та видавання сигналів керування у ті кола машин, які повинні працювати під час виконання цієї операції; у запам'ятовувальних пристроях для розшифрування адреси чи команди, записування або читання коду з певної комірки пам'яті.

Розглянемо роботу простої схеми дешифрування на два розряди. Як і в попередніх випадках, побудуємо її на основі тригерних схем (див. рис. 1.11). Пояснення роботи такого пристрою не потребує особливих зусиль.



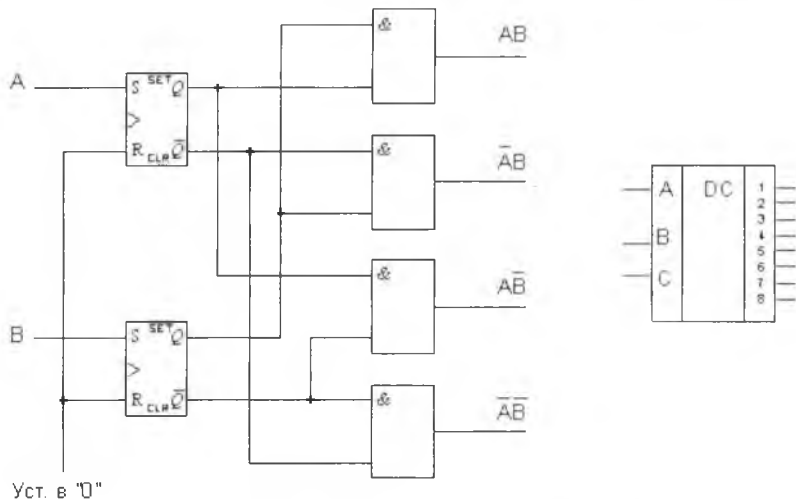


Рис. 1.11. Схема дворозрядного дешифратора  
(з правого боку показано схемне позначення трирозрядного дешифратора)

**Суматор** – пристрій, що виконує сумування чисел на підставі правил порозрядного додавання з урахуванням переносів, які спрямовують у старші розряди.

Суматори класифікують:

- за виглядом елементів, які використовують – комбінаційного та накопичувального типів;
- за способом введення-виведення чисел – паралельної та послідовної дії;
- за способом зображення чисел – двійкові та десяткові;
- за способом організації перенесення – з послідовним та наскрізним перенесенням.

Розглянемо суматори комбінаторного типу (комбінація логічних елементів І, АБО, НЕ).

Сигнал на виході (сума) утворюється тільки в разі визначеної комбінації вхідних сигналів (доданків), які подаються одночасно. Після зникнення вхідних сигналів вихідний сигнал зникає (запам'ятовувальних властивостей немає). Тому такі суматори працюють з регістром, у який записується результат.

Процес сумування і перенесення з сусіднього розряду в однорозрядній сумувальній схемі розбито на дві аналогічні операції: сумування

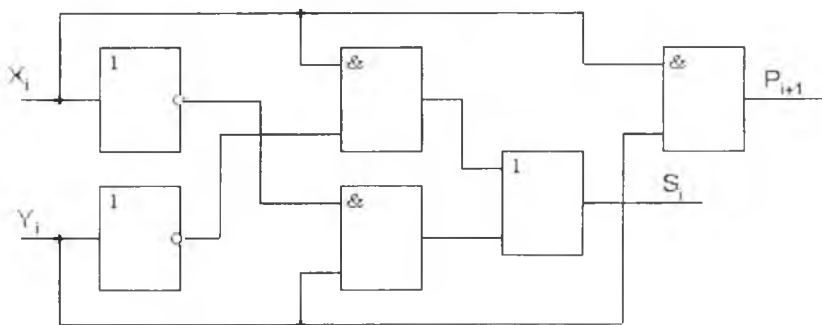
дворозрядних доданків і сумування з отриманим результатом одиниці перенесення. Кожну з функцій виконує схема, яку називають *напівсуматором*.

Розглянемо роботу напівсуматора, яку описують логічними виразами для суми:

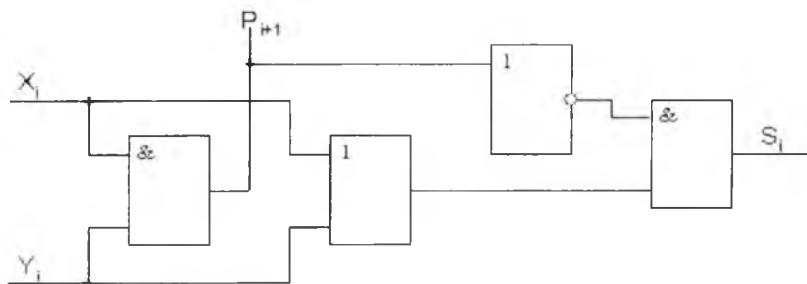
$$\begin{aligned} S_i &= X_i \bar{Y}_i + \bar{X}_i Y_i = X_i \bar{Y}_i + \bar{X}_i Y_i + X_i \bar{X}_i + Y_i \bar{Y}_i = \\ &= X_i (\bar{X}_i + Y_i) + Y_i (\bar{X}_i + \bar{Y}_i) = (X_i + Y_i)(\bar{X}_i + \bar{Y}_i) = \bar{X}_i \bar{Y}_i (X_i + Y_i) \end{aligned}$$

та перенесення  $P_{i+1} = X_i Y_i$ .

Результат для суми отримано внаслідок певних алгебричних перетворень. Для прикладу розглянемо схеми, які реалізують проміжний і кінцевий етапи перетворень (рис. 1.12).



а

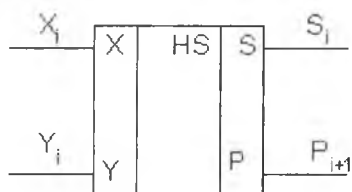


б

Рис. 1.12. Схеми напівсуматорів комбінаційного типу:

а – проміжний, б – кінцевий

На схемах позначають



Якщо додати ще один напівсуматор, то отримаємо схему повного суматора на три входи (рис. 1.13).

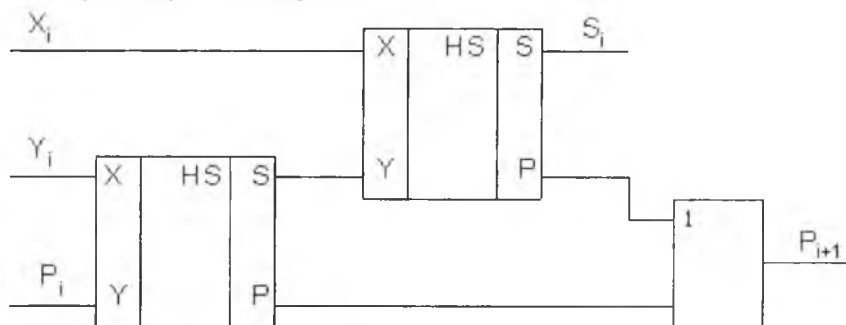
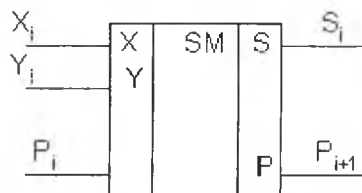


Рис. 1.13. Схема повного суматора на три входи

Повний суматор на схемах позначають так:



Багаторозрядний суматор можна отримати простим послідовним з'єднанням однорозрядних суматорів (рис. 1.14).

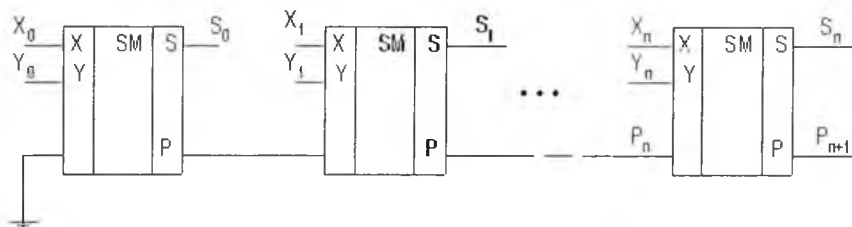


Рис. 1.14. Схема багаторозрядного суматора

Для зменшення часу поширення сигналу перенесення використовують схеми прискореного перенесення.

Однорозрядний суматор можна використати для виконання операції віднімання. Для цього на вхід  $Y$  подають або прямий, або обернений код доданка, який відповідає його від'ємному значенню (рис. 1.15).

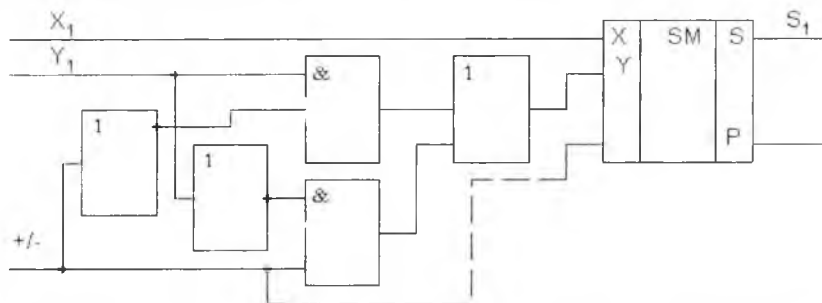


Рис. 1.15. Використання однорозрядного суматора для операції віднімання

Залежно від сигналу керування  $+/-$  на вхід суматора подається прямий код  $Y$  (при  $+/- = 0$ ) або обернений код  $Y$  (при  $+/- = 1$ ). Для формування додаткового коду доданка до його інверсного коду треба додати 1 молодшого розряду. Цього досягають з'єднанням входу керування  $+/-$  із входом суматора молодшого розряду.

Трохи складніше, однак можна зібрати з наведених вище логічних схем і вузлів пристрої для виконання операції множення та ділення двійкових чи десяткових чисел.

### Контрольні запитання

1. Яка відмінність між термінами ЕОМ, ПЕОМ і ПК?
2. Що ми називаємо інформацією?
3. Дайте визначення біта.
4. Назвіть три головні функції алгебри логіки.
5. Запишіть закони інверсії.
6. Що таке тригер і які типи тригерів ви знаєте?
7. Які стани дозволені, а які заборонені у RS-тригері?
8. Які функції регістрів і які типи регістрів використовують у побудові ЕОМ?
9. Як побудувати десятковий лічильник?
10. Для чого використовують шифратори та дешифратори?
11. Які типи суматорів ви знаєте?

## 1.4. Класифікація архітектур ОС

*Архітектура ЕОМ* (будова, внутрішнє облаштування) є збірним поняттям і містить у собі логічну організацію ОС, способи подання і формати даних, набір команд, організацію пам'яті і системи адресації, організацію системи введення-виведення даних та ін. Архітектуру ЕОМ можна розглядати як певну сукупність апаратних і програмних засобів з погляду програміста, а точніше системного програміста, який розробляє програми на машинозалежній мові.

Для ЕОМ першого і другого покоління характерною була класифікація за системою команд. Практично перехід від однієї системи до іншої означав повну перекваліфікацію програміста. У ЕОМ третього і четвертого покоління щораз більше застосовували мови програмування високого рівня (Фортран, ПЛІ, Алгол та ін.), і програмісти головню не потребували знання системи команд, структури пам'яті, способів адресування, розрядності регістрів тощо. Однак для системного програміста і сьогодні актуальними є питання структури ЕОМ, ємності пам'яті і взаємодії між компонентами, особливо між процесором і периферією, тому, погоджуючись з А. Смирновим [23], розглянемо таку класифікацію.

### 1.4.1. Класифікація за інтегральними ознаками

Доцільно скористатися поняттям архітектури ЕОМ як певним набором інтегральних характеристик, а саме:

- сукупність властивостей і характеристик, що визначають інформаційні зв'язки між центральним процесором (ЦП), периферійними процесорами (ПП), оперативною пам'яттю (ОП), зовнішньою пам'яттю і периферією;
- взаємодією і структурою потоків команд та потоків даних.

Перша характеристика дає змогу врахувати такі особливості ЕОМ третього і четвертого поколінь, як багатопроцесорність, моношінна структура зв'язків, швидкодія, розподілена пам'ять.

#### 1.4.1.1. Класифікація за взаємодією ЦП, ОП, ПП

Розглянемо інформаційні зв'язки між трьома головними типами пристроїв ЕОМ, обмежившись лише структурними зв'язками в архітектурі ОС. Є такі структури:

- однопроцесорна (з одним ЦП і периферійним процесором чи каналом) зі зв'язком через ЦП;

- з одним потужним ЦП і кількома периферійними процесорами, що його обслуговують;
- багатопроцесорна (кілька ЦП, ПП і секціонована ОП);
- з магістральною шиною;
- мережна;
- функціонально переналагоджувальна багатопроцесорна;
- мас-процесорна (значна кількість процесорів).

В однопроцесорній структурі ЦП (спільно з операційною системою) координує всю роботу системи. Потрібні прямі зв'язки з периферійними пристроями ЦП виконує через канали або ПП (рис. 1.16). Така структура була характерною для ЕОМ третього покоління: IBM 360/370, ЕС ЕОМ, ICL System4 та інші. Ця архітектура застосовується також у сучасних пристроях, які забезпечують виконання спеціальних функцій у комп'ютерних мережах.

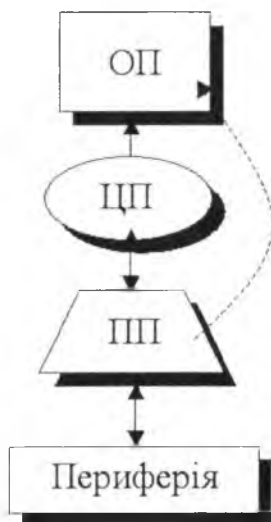


Рис. 1.16. Однопроцесорна архітектура ЕОМ

Однак для продуктивніших систем такі структури, коли при кожному звертанні необхідно переривати роботу ЦП, не вигідні. Тому виникла ідея поєднати роботу потужного ЦП із кількома менш потужними, функція яких полягає в обслуговуванні переривань від зовнішніх пристроїв, а також ОЗП. Наприклад, це може бути структура з концентричним розташуванням периферійних процесорів (див. рис. 1.17).

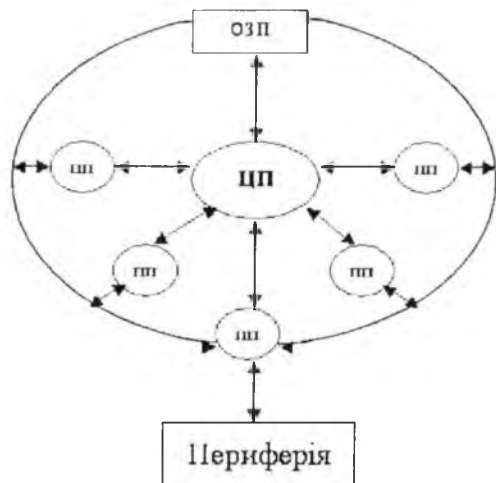


Рис. 1.17. Архітектура з периферійними процесорами

Подібну структуру мають високопродуктивні системи CDC. А от ЕОМ “Ельбрус” (кількість ЦП від 1 до 10, ПП від 1 до 4, кількість секцій ОП від 1 до 8), яку донедавна використовували тільки для військових потреб, була побудована за принципом збільшення кількості ЦП з однією ОП. Система може працювати за алгоритмом розпаралелювання задачі на кілька процесів або розв’язувати на кожному з них свою задачу. Ця система застосована також у Burroughs-700 (США), Iris-80 (Франція) та ін. (рис. 1.18).

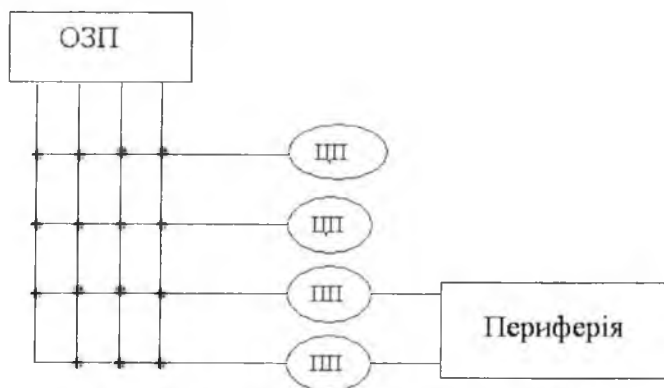


Рис. 1.18. Багатопроцесорна архітектура ЕОМ

Збільшення продуктивності такої системи близьке до лінійного до три-процесорного варіанта (рис. 1.19). Тому серед ОС високої продуктивності популярні дво- і чотирипроцесорні системи (Cray-X-MP; IBM "С'єрра" та ін.). Завдяки використанню комплексу додаткових засобів і архітектурних знахідок з опорою на кеш-пам'ять великої потужності вдалося отримати майже лінійне зростання продуктивності зі збільшенням кількості процесорів до п'яти-десяти для широкого класу задач на ОС родини "Ельбрус".

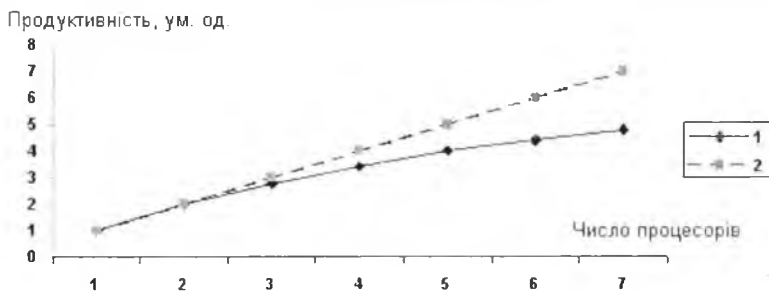


Рис. 1.19. Залежність продуктивності ОС від числа процесорів:  
1 – реальна, 2 – очікувана

Нові можливості відкриваються для паралельних ОС, якщо звернутися до різного типу сіткових алгоритмів. Такими алгоритмами розв'язують багато задач неперервного середовища, які описуються системами рівнянь у частинних похідних, лінійними системами алгебричних рівнянь. Процесори, які опрацьовують інформацію та обмінюються інформацією між собою, розміщені у вузлах решітки або сітки. Таких процесорів у ОС міститься вже декілька сотень або навіть тисяч. Основою ОС стають мікропроцесори. На відміну від багатопроцесорних систем з великими універсальними процесорами, такі системи називають *мас-процесорними*.

Операції розподіляють за окремими функціональними пристроями, які працюють паралельно, проте в самих функціональних пристроях застосована конвеєризація (див. 1.4.3), а також відбувається попередній перегляд програм, а їхньому швидкому попередньому і паралельному виклику з ОП допомагає глибоке розшарування пам'яті. Детальніше про це див. напр., у [23].

Розрізняють такі мас-процесорні ОС:

- багатопроцесорні скалярні (IBM 3090/200, EC 1065);
- векторні і векторно-конвеєрні (CDC 6600, Cray-I);
- мас-процесорні з динамічною архітектурою;
- спеціальні процесорні.



ОС з *магістральною шиною* детальніше розглядаються у розділі 2. Тут зазначимо лише таке: магістрально-шинна структура передбачає, що всі частини ОС можуть бути приєднаними до однієї багатопровідної шини (або коаксіального чи оптоволоконного кабелю), захоплюючи її на короткий період машинного такту для обміну інформацією (рис. 1.20). Конфліктну ситуацію вирішує арбітр шини.

Розрізняють шину адреси, шину даних та шину керування.



Рис. 1.20. Приклад архітектури ОС з магістральною шиною

Окремий напрям становить мережна архітектура ОС. Розрізняють глобальні, регіональні та локальні комп'ютерні мережі. Вони мають свою топологію і їхня робота регламентована певним набором правил (протоколів). Про це детальніше див. у розділі 3.4.

#### 1.4.1.2. Класифікація за взаємодією потоку команд і потоку даних (класифікація М.Флінна)

Розглянемо дещо іншу класифікацію, яка властива головно супер-ЕОМ. Відомо, що в ЕОМ з класичною архітектурою опрацювання даних є послідовним. Команди надходять одна за одною, а для них з ОП чи з регістрів так само послідовно надходять операнди. Одній команді відповідає один необхідний для неї набір операндів; для бінарних операцій, як звичайно – два набори. Скорочено такий тип архітектури називають ОКОД (одна команда – одні дані); англійська аббревіатура SISD (Single Instruction, Single Data) (рис. 1.21).



Рис. 1.21. Тип архітектури ОКОД

Для спеціальних процесорів, що опрацьовують зображення, або асоціативних процесорів опрацьовується набір з множини даних. Це тип ОКБД (одна команда – багато даних); англійська аббревіатура SIMD (Single Instruction, Multiple Data) (рис. 1.22). Кількість елементів, що опрацьовуються, може бути дуже великою (тисячі), проте їх опрацювання відбувається синхронно (приклад: сучасні процесори з технологією MMX).

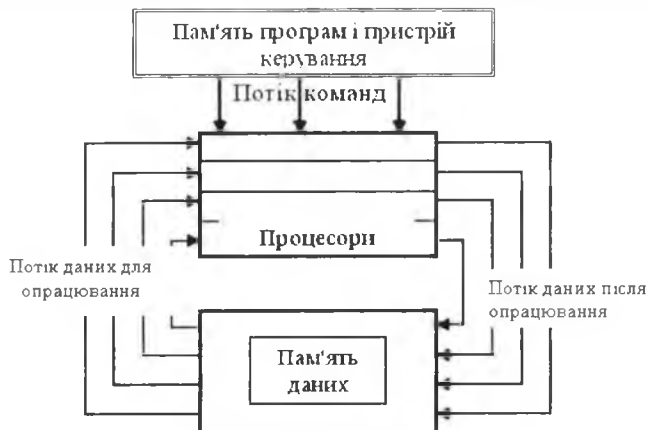


Рис. 1.22. Тип архітектури ОКБД

Прикладом типу БКОД (багато команд – одні дані) може бути конвеєр, тобто послідовне опрацювання одного потоку даних багатьма опрацьовувальними ступенями конвеєра (рис. 1.23).

#### Конвеєр процесорів (магістраль опрацювання)



Рис. 1.23. Тип архітектури БКОД

Тип БКБД (багато команд – багато даних) передбачає найповніше і незалежне розпаралелювання процесу (рис. 1.24).

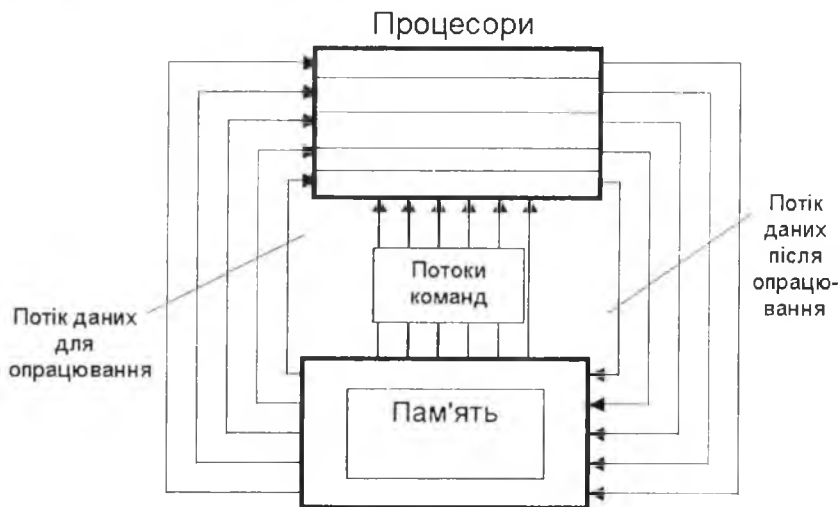


Рис. 1.24. Архітектура БКБД

Розглянута вище класифікація є в багатьох працях, і хоча вона й відображає головну взаємодію потоків даних і команд, однак все ж не охолює всіх можливих взаємодій, які трапляються на практиці.

Наприклад, проаналізуємо конвексно-векторне опрацювання. Щодо потоку даних та інтегрального опрацювання дії – це тип ОКБД. Проте опрацьовувальну дію можуть виконувати підконвекси, тобто множина послідовно виконуваних команд над кожною парою операндів, а це тип БКОД.

Існують й інші класифікації архітектур, які дозволяють краще розрізняти ОС за принципами множинності (систематика Р. Хокні та К. Джессхоупа) чи систематика Ф.Шара. Однак життя, як звичайно, вносить свої корективи навіть у найретельніше відпрацьовані схеми і класифікації. Тому конструктори ЕОМ прагнуть досягти не найгармонійнішої одноманітної архітектури, а максимальної продуктивності в заданому класі задач чи найліпшого співвідношення продуктивності ОС до її вартості, застосовуючи всі можливі архітектурні й апаратні хитрощі [9, 16, 23, 25].

### 1.4.2. Класифікація за функціональним призначенням

Як уже зазначено, сучасні обчислювальні системи умовно поділяють на аналогові, дискретні та гібридні. Кожна з цих груп має ще свій внутрішній поділ, який виглядає приблизно так [1]:



### 1.4.3. Класифікація за способом організації виконання команд

CISC (Complete Instruction Set Computer) – комп'ютери з повним набором інструкцій (команд), які виконує мікропроцесор. До цього типу належить сімейство ПЕОМ на базі мікропроцесорів x86. Унаслідок широкого набору команд архітектура таких процесорів страждає від надміру апаратних ресурсів, склад і призначення регістрів досить неоднорідні. На виконання команд необхідно витратити багато часу, причому час виконання різних команд неоднаковий (різна кількість машинних тактів). Ці фактори негативно впливають на продуктивність ПЕОМ, тому починаючи з МП Intel486 застосовують комбіновану архітектуру, а саме у CISC-процесор вкраплюють RISC-ядро.

МП з архітектурою **RISC** (Reduced Instruction Set Computer) використовують порівняно невеликий (скорочений) набір найуживаніших команд. Цей набір визначено за результатами статистичного аналізу значної кількості програм для головних галузей застосування CISC-процесорів.

Особливістю RISC-архітектури є те, що всі команди мають однаковий формат і працюють з операндами, які розташовані у регістрах процесора. Звертання до пам'яті відбувається за допомогою спеціальних команд завантаження регістра і запису.

Невеликий набір команд і простота їхньої структури дають змогу повністю реалізувати апаратне виконання й ефективний конвеєр, використовуючи порівняно невелику кількість обладнання. Завдяки цьому команди виконуються в 2-4 рази швидше, ніж у звичайних CISC з тією ж тактовою частотою. Продуктивність такого процесора на 30% більша від звичайного.

Перші експериментальні МП (RISC II, MIPS, IBM 801) з'явилися ще у 1980-81 рр. Мікропроцесорну революцію продовжили провідні фірми у 1985-86 рр. (Acorn, AMD, IBM, Sun Microsystems). Переважно це тримікронна CMOS-технологія, розрядність шини даних 32 розряди.

Сьогодні RISC-процесори здебільшого використовують для побудови співпроцесорів та спецпроцесорів, інтелектуальних контролерів, багатопроцесорних систем та ін. Кількість команд для різного типу МП RISC різна, наприклад, МП фірми Acorn – 26 команд, IBM – 118 команд.

Для прикладу покажемо, якого виграшу можна досягнути у МП I860 з ядром RISC, порівняно зі звичайним послідовним виконанням команд. I860 – це 64-бітовий МП, швидкодія якого сягає 80 Mflops (Million of Floating-point Operations Per Second) і 40 MIPS (Million of Instructions Per Second) на тактовій частоті 40 МГц. Процес виконання команд розбито на чотири етапи:

<b>F</b>	<b>D</b>	<b>X</b>	<b>W</b>
Вибирання	Декодування	Виконання	Запис

Однотипність виконання команд дозволяє організувати їх конвеєрне опрацювання. Тут варто дати деякі пояснення.

Конвеєр – пристрій, у якому одночасно кілька команд програми проходять різні стадії обробки. У класичному 5-рівневому конвеєрі п'ять команд одночасно проходять такі стадії виконання команди:

- читання [вибирання] команди з буфера (з кешу L1, L2);
- дешифрування команди;
- читання операндів;
- виконання команди;
- запис результатів виконання команди.

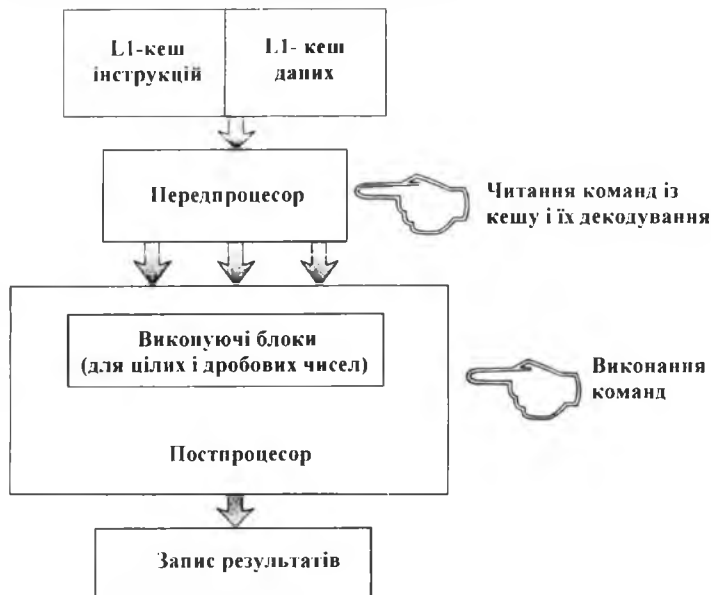
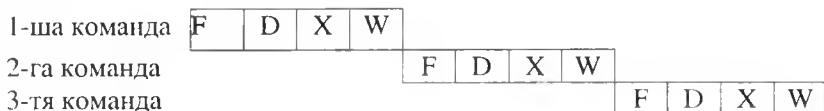


Рис. 1.25. Класичний конвеєр

Використання конвеєра дозволило підвищити швидкодію процесора. У сучасних мікропроцесорах кількість “сходинок” конвеєра значно більша (операції читання команди з буфера, дешифрування команди, виконання команди, запису результатів виконання команди розділені на дрібніші етапи).

У випадку скалярного виконання команд МП І860 послідовність виконання матиме такий вигляд:



Під час конвеєрного виконання за той самий час виконається утрічі більше команд:

1-ша команда	F	D	X	W							
2-га команда		F	D	X	W						
3-тя команда			F	D	X	W					
4-та команда				F	D	X	W				
5-та команда					F	D	X	W			
6-та команда						F	D	X	W		
7-ма команда							F	D	X	W	
8-ма команда								F	D	X	W
9-та команда									F	D	X
10-та команда										F	D
11-та команда											F

Ще один тип комп'ютерів – з MISC (Minimise Instruction Set Computer) архітектурою. Термін MISC був запропонований Чаком Муром. Ідея цієї архітектури полягає в тому, що скорочення числа команд до оптимального мінімуму покращить продуктивність процесора і зменшить вартість побудови комп'ютера. За такою схемою були сконструйовані ЕОМ “Сетунь” (24 команди) та ряд ЕОМ “Forth”. Наприклад, мікропроцесор F21 послуговується тільки 27-а 5-розрядними командами. Складніші команди формують як послідовність простих. Процесор має стекову архітектуру і досягає пікової швидкодії понад 100 MIPS. Апаратна простота MISC процесорів має значні переваги у порівнянні з RISC чи CISC процесорами в категорії “продуктивність/вартість” і має меншу споживану потужність.

## 1.5. Типи даних ПЕОМ

Для всіх сучасних МП базовою одиницею даних є байт, тобто 8-бітова величина. Після того, як дані локалізовано у пам'яті процесора, МП може інтерпретувати їх різними способами, залежно від змісту команди чи намірів програміста. Для кодування інформації використовують в основному двійкову, десяткову та шістнадцяткову системи числення. У сучасних МП реалізовано апаратну підтримку багатьох типів даних, таких як числа, вказівники (pointer) та рядки (string). Тридцятидворозрядні мікропроцесори підтримують усі типи даних, які використовуються у мовах високого рівня:

1. *Біт (bit)* – поодинокий двійковий розряд.
  2. *Бітове поле (bit field)* – група, що охоплює до 32 суміжних бітів, що займають максимально до 4 байтів.
  3. *Рядок бітів (bit string)* – послідовність бітів довжиною до 4 Гбіт.
  4. *Байт (byte)* – ціле число, що займає 8 бітів (старший розряд відводиться під знак).
  5. *Беззнаковий байт (unsigned byte)* – натуральне число, що займає 8 бітів.
  6. *Ціле слово (integer word)* – ціле число, що займає 16 бітів.
  7. *Беззнакове ціле (unsigned integer, word)* – натуральне число, що займає 16 бітів.
  8. *Довге ціле (long integer, long word)* – ціле число, що займає 32 біти.
  9. *Довге беззнакове ціле (unsigned long integer)* – натуральне число, що займає 32 біти.
  10. *Чотирикратне ціле (quad integer, quad word)* – ціле число, що займає 64 біти.
  11. *Чотирикратне беззнакове ціле (unsigned quad integer)* – натуральне число, що займає 64 біти. (Зауважимо, що в усіх числах зі знаком старший біт є знаковим. Як і у випадку 16-розрядних процесорів для позначення додатнього числа біт дорівнює 0, а від'ємного – 1).
  12. *Символ (char)* – байтове подання керівних та алфавітно-цифрових символів ASCII.
  13. *Рядок (string)* – неперервна послідовність, утворена із байтів слів чи подвоєних слів від 1 байта до 4 Гбайт.
  14. *Двійково-десятковий код (binary coded decimal, BCD)* – байтове (незапаковане) подання десяткових цифр від 0 до 9.
  15. *Запакований двійково-десятковий код* – байтове подання від 0 до 9 в одному байті.
  16. *Короткий вказівник, зсув (short pointer, offset)* – 16- або 32-розрядна адреса, що непрямо визначає комірку пам'яті.
  17. *Вказівник (pointer)* – повна адреса елемента пам'яті, яка містить 16-бітний або 32-розрядний зсув.
- У тих випадках, коли МП взаємодіє з арифметичним співпроцесором, додатково використовують знакові 32-, 64- та 80- розрядні форми даних з плаваючою комою.



Арифметичні типи даних ПЕОМ, побудованих на мікропроцесорах Intelx86

Тип даних	Біти	Кількість десяткових значущих цифр	Діапазон значень
Коротке ціле	16	4	$-32768 \leq x \leq +32767$
Ціле	32	9	$-2 \cdot 10^9 \leq x \leq +2 \cdot 10^9$
Довге ціле	64	18	$-9 \cdot 10^{18} \leq x \leq +9 \cdot 10^{18}$
Запаковане десяткове	80	18	$-99..99 \leq x \leq +99..99$
Коротке дійсне	32	6-7	$8,43 \cdot 10^{-37} \leq  x  \leq 3,37 \cdot 10^{38}$
Подвійне дійсне	64	15-16	$4,19 \cdot 10^{-307} \leq  x  \leq +1,67 \cdot 10^{308}$
Довге подвійне дійсне	80	19	$3,4 \cdot 10^{-4932} \leq  x  \leq +1,2 \cdot 10^{4932}$

### Пояснення запакованого та розпакованого формату:

а) У розпакованому форматі кожний байт містить одну десяткову цифру, яка зображується у молодшій тетраді. Знак числа кодують у різних ЕОМ по-своєму. У ПЕОМ знак десяткового числа кодують у крайньому лівому байті, причому знак “-” кодується 1, знак “+” – кодується 0.

Наприклад, число -567 у розпакованому форматі буде мати вигляд:

-	5	6	7
11111111	00000101	00000110	00000111
Знак	Цифра		

У неупакованому форматі цифри, як і символи, зображають у певній кодовій таблиці (ASCII, Windows 1251, CP866, KOI8 – один байт, 255 символів). Розрізняють дві частини таблиці кодів: 1 – основна, 2 – додаткова (з урахуванням національних стандартів):

- 1) 0-127
- 2) 128-255.

Для більшої кількості символів у кодовій таблиці використовують кодову таблицю Unicode (два байти, 65535 символів).

Десяткові числа від 0 до 9 у кодуванні ASCII відповідають шістнадцятковим кодам від 30h до 39h відповідно.

б) В упакованому форматі десяткове число займає половину байта, тобто у кожному байті міститься дві десяткові цифри з вагами розрядів 8421 у кожній тетраді. Діапазон чисел, який може бути представлений в одному байті, знаходиться у межах від 0 до 99.

Наприклад, число +963,8 в упакованому форматі буде таким:

+	9	6	3	,	8
00000000	1001	0110	0011		1000
Знак			Цифра		

Положення десяткової коми може бути довільним, зокрема для ПЕОМ вона знаходиться після третього зліва розряду числа.

### Контрольні запитання

1. У чому полягає суть класифікації за інтегральними ознаками?
2. Яка залежність продуктивності ЕОМ від кількості процесорів?
3. Наведіть приклади мас-процесорних систем.
4. Які типи взаємодії потоку команд і потоку даних?
5. Як класифікують обчислювальні системи за функціональним призначенням?
6. Якими типами даних оперує мікропроцесор?
7. Яка відмінність між запакованим і розпакованим форматом чисел?
8. Яка розрядність чисел з плаваючою комою у ПЕОМ?
9. Яка відмінність між CISK- і RISK-процесорами?
10. Які реалізації MISC-архітектури Ви знаєте?

## Розділ II. ЗАГАЛЬНІ ПРИНЦИПИ ПОБУДОВИ ПЕОМ

Архітектура ЕОМ залежить від типу центрального процесора чи мікропроцесора (МП). Визначальним моментом є розрядність МП (8, 16, 32, 64 тощо). ЕОМ, зібрані на базі цих МП, мають не тільки певний тип МП, а й функціональні розширювачі, контролери, гнучкі та жорсткі диски, клавіатуру, дисплей та інші вузли.

Відомо також, що розв'язування певної задачі на ЕОМ поділене на частини. Найпростіша частина – це машинна команда. Набір машинних команд, які записані у певній послідовності, становлять машинну програму. Для реалізації тієї чи іншої програми необхідні арифметико-логічний пристрій, пристрій пам'яті, пристрій керування, пристрій введення/виведення інформації.

Взаємодія цих пристроїв відбувається через так звану системну магістраль (СМ), або шину. У переважній більшості випадків СМ – це розширення шини МП, однак, не завжди (напр., МП може бути 16-розрядним, а зовнішня шина – 8-розрядною.)

Незалежно від архітектурного виконання структурну організацію взаємодії компонентів ПЕОМ можна зобразити як на рис. 2.1.

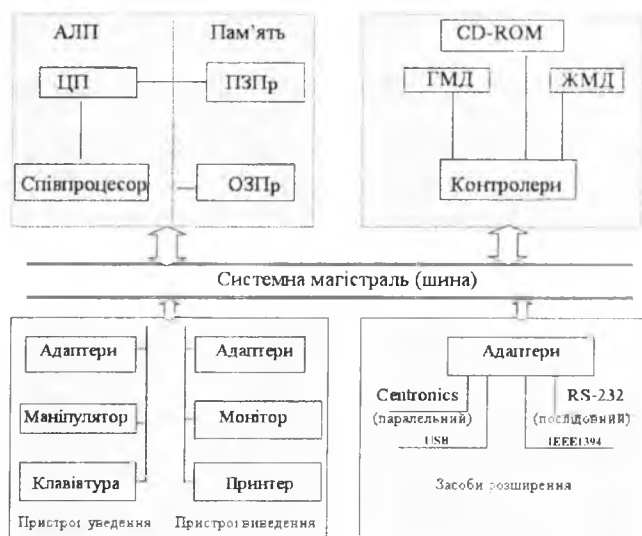


Рис 2.1. Структурна схема сучасної ПЕОМ (пояснення скорочень див. далі)

Структурна схема може дещо видозмінюватися залежно від призначення ЕОМ: побутова, професійна чи спеціалізована.

Між МП та оперативною пам'яттю (ОЗПр) є зв'язок, тобто розрядність МП пов'язана з обсягом ОЗПр. Наприклад, 8-розрядному МП відповідає 64 Кбайт ОЗПр, 16-розрядному – від 1 до 16 Мбайт, 32-розрядному – до 4 Гбайт.

Швидкість виконання операцій визначається тактовою частотою МП та магістральної шини. Прості операції додавання цілих чисел типу “пам'ять – пам'ять” виконуються за 20-30 тактів, а множення-ділення – за 130-150. Знаючи тактову частоту процесора, легко визначити його швидкодію.

## **2.1. IBM-подібні ПЕОМ**

### **2.1.1. Програмна архітектура IBM-подібних ПЕОМ**

Програмну архітектуру ПЕОМ визначає, насамперед, тип базової архітектури мікропроцесора. Це структура регістрів, формати команд і даних, адресування даних і операндів. Іншим важливим моментом у реалізації програмної архітектури є організація та способи керування оперативною пам'яттю. Механізми сегментної та сторінкової організації пам'яті діють роздільно або сумісно. Система команд цілочислового пристрою та пристрою з плаваючою комою мабуть найповніше відображає всі можливості ПЕОМ для програміста чи просто користувача. Тут враховані можливості регістрової архітектури, механізмів адресування, захисту та привілеїв, організації віртуальної пам'яті, налагодження та ін. Особливістю програмної архітектури є також реагування на переривання роботи процесора. До програмних особливостей архітектури належить ініціалізація процесора, контроль за роботою кеш-пам'яті, режим віртуального процесора та ін.

#### **2.1.1.1. Регістрова структура 16-розрядних МП фірми Intel**

Регістрова структура МП покликана забезпечити виконання всіх його функцій. Розглянемо спочатку реалізацію можливостей регістрової архітектури у МП Intel 8086 (8088), яка на сьогодні являє лише історичний інтерес. Структурна схема цього процесора зображена на рис. 2.30. МП містить п'ятнадцять 16-розрядних та шість 8-розрядних регістрів, які розподілені у пристроях спряження з магістраллю та опрацювання команд.

У складі пристрою спряження з магістраллю є:

- шість 8-розрядних регістрів черги команд (РЧК);
- чотири 16-розрядні сегментні регістри (C,D,S,E);
- 16-розрядний регістр адреси команд (IP);
- 16-розрядний регістр обміну (OP).

Регістр черги команд влаштований так, що команди виконуються за правилом “перший прийшов — першим обслужили”. У МП 8088 черга команд у РЧК скорочена до чотирьох.

Сегментні регістри відповідають чотирьом сегментам пам’яті:

- DS – даних;
- SS – стеку (стек – спеціально організована область ОЗП, яку використовує ЕОМ для тимчасового зберігання даних або адрес. Число, записане в стек останнім, опрацьовується першим);
- CS – коду;
- ES – проміжних даних.

У кожному з цих регістрів зберігається 16 старших розрядів коду адреси відповідного сегмента пам’яті.

**Пристрій опрацювання даних** містить вісім 16-розрядних регістрів загального призначення, регістр обміну та регістр ознак.

**Програмно-доступними** функціональними частинами є **регістри загального призначення** (для зберігання операндів та результатів виконання команд), **сегментні регістри** (для зберігання базових адрес поточних сегментів пам’яті), **адреси команд і ознак**.

Регістри загального призначення розбиті на дві групи по чотири у кожній:

- 1) даних;
- 2) індексні і вказівники.

Старші і молодші вісім розрядів групи регістрів можуть бути адресовані окремо. В такому випадку вони утворюють набір з восьми 8-розрядних регістрів. Регістри даних можна без обмеження використовувати у більшості арифметичних і логічних операцій (рис. 2.6).

Інша група регістрів – вказівники бази і стеку та два індексні регістри (джерела і приймачі) – також може брати участь у більшості арифметичних і логічних операцій. Крім того, деякі команди використовують ці регістри неявно у таких випадках:

- множення, ділення і введення-виведення слів (AX);
- множення, ділення і введення-виведення байтів, перекодування, десяткова арифметика (AL);

- множення, ділення байтів (AH);
- перекодування (BX);
- операції з рядками, цикли (CX);
- множення і ділення слів, неявне введення-виведення (DX);
- операції зі стеком (SP);
- операції з рядками (SI, DI).

У регістрі ознак (рис. 2.2) використано тільки дев'ять розрядів. Шість із них слугують для реєстрації особливостей результату арифметичних і логічних операцій:

- ознака знака SF, якщо знаковий біт результату операції = 1;
- ознака нуля ZF, якщо результат операції дорівнює нулю;
- ознака додаткового переносу за наявності перенесення з третього розряду AF (використовують команди десяткової арифметики);
- ознака парності PF, якщо результат містить парну кількість одиниць;
- ознака перенесення CF за наявності перенесення (у разі додавання) або позички (у разі віднімання) зі старшого розряду результату. Цю ознаку використовують команди додавання і віднімання багатобайтових чисел. Команди циклічного зсуву можуть ізолювати розряд, що зсувається у пам'яті чи регістрі, шляхом розміщення його у розряді ознаки перенесення;
- ознака OVERFLOW (OF) відображає переповнення у разі виконання операцій з числами зі знаком. Вона ініціює внутрішнє переривання.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				O	D	I	T	S	Z		A		P		C
				F	F	F	F	F	F		F		F		F

Рис. 2.2. Склад регістра ознак МП 8086(88)

У регістрі ознак є ще три розряди, які можна використати для керування мікропроцесором:

- ознака DF = 1 – викликає автодекремент під час виконання операцій з рядками даних. Це означає, що опрацювання рядків відбувається від старших адрес до молодших (справа наліво). DF = 0 викликає автоінкремент, тобто опрацювання рядків зліва направо.
- ознака IF = 1 — дозвіл на переривання. Мікропроцесор реагує на зовнішні масковані запити переривань.

IF = 0 — заборона переривання. У цьому разі IF не впливає на внутрішні переривання і немасковані зовнішні;

- ознака TF = 1 — переводить мікропроцесор у покроковий режим роботи. У цьому режимі після кожної команди мікропроцесор автоматично генерує переривання.

Прямого зв'язку ПО із зовнішньою магистраллю немає, а обмін даними з ПСМ відбувається через регістр обміну (PO).

Інший 16-розрядний процесор – 80286 за набором програмно доступних регістрів є таким самим, за винятком деяких змін у регістрі ознак. У цьому МП додано також регістр стану машини та регістри глобальної та локальної таблиць дескрипторів, регістр дескриптора таблиці переривань, регістр задачі. Про призначення цих регістрів див. пп. 2.2.1.3.

### 2.1.1.2. Регістрова структура 32-розрядних МП фірми Intel

Процесори, починаючи з 80386, мають 32 програмно-доступних регістри у таких категоріях:

- 1) регістри загального призначення;
- 2) регістри сегментування;
- 3) регістр ознак;
- 4) регістри керування;
- 5) регістри системної адреси;
- 6) регістр тестів;
- 7) регістр відлагоджування.

Ці регістри існували також у 8086, 80286, тому всі 16-розрядні регістри всіх процесорів попередніх випусків “покриваються” 32-розрядними регістрами.

31	16	15	0
		AX	EAX
		BX	EBX
		CX	ECX
		DX	EDX
		SI	ESI
		DI	EDI
		BP	EBP
		SP	ESP

Рис. 2.3. Регістри даних і адресування

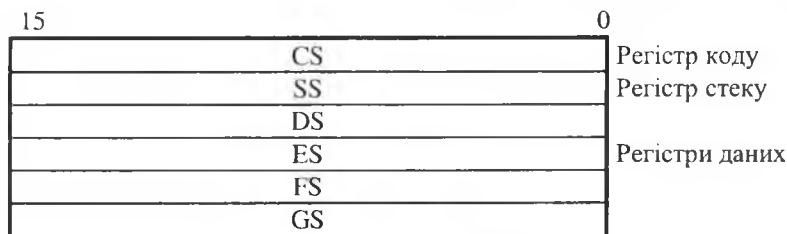


Рис. 2.4. Сегментні регістри

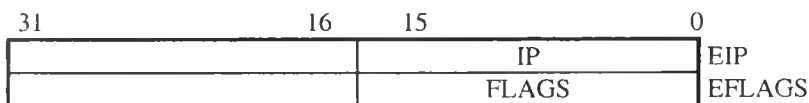


Рис. 2.5. Регістри вказівника команд і ознак

На рис. 2.3–2.5 показані всі регістри базової архітектури мікропроцесора, які містять регістри загальної адреси і даних, вказівник команд, регістр ознак. Ці регістри у разі включення нової задачі завантажуються новим вмістом.

Базова архітектура містить шість прямо доступних сегментів, кожний розміром до 4 Гбайт. Сегменти відображені значеннями селекторів, поміщеними в регістр сегмента.

Селектори автоматично завантажуються під час операції включення задач.

Інший тип регістрів – це регістри керування, системної адреси, відлагоджування і тестів. Їх використовують для налагодження операційної системи.

**Регістри загального призначення (РЗП).** Мікропроцесор має вісім 32-розрядних РЗП. Вони підтримують 16-, 32-розрядні адресні операнди, 1, 8, 16, 32, 64 операнди даних і бітові поля від 1 до 32 бітів. Їх називають: EAX, EBX, ECX, EDX, ESI, EDI, EBP, ESP.

Молодші 16 розрядів кожного регістра можна використовувати окремо, це AX, BX, CX, DX, SI, DI, BP, SP. Також можна індивідуально застосовувати молодший (0-7) і старший (8-15) байти РЗП AX, BX, CX, DX: їх називають, відповідно, AL, BL, CL, DL та AH, BH, CH, DH (див. рис. 2.6).



31	16 15	8 7	0
EAX	AH	A	AL
EBX	BH	B	BL
ECX	CH	C	CL
EDX	DH	D	DL
ESI	SI		
EDI	DI		
EBP	BP		
ESP	SP		

Рис. 2.6. Регістри загального призначення

Індивідуальна доступність байтів РЗП забезпечує додаткову гнучкість для операцій з даними, однак цього не використовують під час адресування.

**Вказівник команд (IP).** Регістр вказівника команд (див. рис. 2.5) є 32-розрядним, його називають EIP. Він містить зміщення наступної команди, яку потрібно виконати. Зміщення завжди визначене щодо сегмента коду. Його молодші 16 біт називають IP і використовують для 16-розрядного адресування.

**Регістр ознак** є 32-розрядним і називається EFLAGS. Частина бітів EFLAGS, як показано на рис. 2.7, керують певними операціями і відображають статус 32-розрядного процесора. Молодші 16 розрядів EFLAGS становлять регістр FLAGS, який можна використовувати під час виконання операцій з процесором 8086.



Рис. 2.7. Регістр ознак

Найліпший спосіб вивчення регістра ознак — опис його побітно. Структура регістра ознак показана на рис. 2.7, звідки легко побачити, що не кожний біт є визначеним. Невизначені біти є зарезервованими, тобто в цей момент не мають значення, однак можуть бути використані для спеціальних цілей у наступних версіях мікропроцесора.

Усі ознаки молодшого байта регістра задані арифметичними або логічними операціями процесора. За винятком ознаки переповнення, всі ознаки старшого байта молодшого слова відображають стан мікропроцесора і впливають на характер виконання програм.

*Ознаку перенесення CF* використовують для обчислень підвищеної точності, логічних операцій та інших команд. Вона відображає, було перенесення зі старшого розряду операнда чи ні, і, відповідно, становить 1 або 0.

*Ознака парності PF* позначає, парна чи непарна кількість одиниць молодших 8 біт результату. Вона дорівнює 1, якщо результат операції має парну кількість одиниць, інакше — 0.

*Ознака додаткового перенесення AF* дає змогу мікропроцесору виконувати команди десяткової арифметики. Вона відображає, правильно виконалася команда десяткової арифметики чи ні, і, відповідно, дорівнює 0 або 1.

*Ознака нуля ZF* відображає, що результат останньої операції дорівнював нулю.

*Ознака знака SF* показує, додатним чи від'ємним був результат останньої арифметичної або логічної операції.

*Ознака спеціального переривання TF* допомагає відлагоджувати програми. Ця ознака не є наслідком роботи мікропроцесора, її задає програма за допомогою спеціальної команди. Інша назва — *ознака трасування*, або *покрокової роботи*. Якщо вона задана, то після виконання кожної команди виникає переривання по трасуванню. Під час процедури переривання мікропроцесор скидає ознаку трасування. Це дає змогу опрацьовувати переривання по трасуванню без переривання після кожної команди. Після стандартного опрацювання ознака набуває початкового стану, тобто після виконання наступної команди програми знову виникає переривання по трасуванню. Цей процес буде тривати доти, доки програма користувача не скине ознаки TF або не завершиться виконання задачі.

*Ознака переривань* IF керує зовнішніми перериваннями. Доки ознака переривань дорівнює 0, жодні зовнішні переривання мікропроцесор не опрацьовуватиме (за винятком немаскованих). Коли вона дорівнює 1, то опрацьовуватимуться будь-які переривання, що виникають.

*Ознаку напрямку* DF мікропроцесор використовує під час роботи з рядками, які працюють з великими блоками даних, для визначення напрямку просування по блоку, тобто в бік збільшення або зменшення адреси (інкремент/декремент). Якщо ознака дорівнює 1, то команди опрацьовують рядки у бік зменшення адреси, якщо ж ознака дорівнює 0 – то в бік збільшення. Розглянуті біти EFLAGS повністю збігаються з бітами регістра FLAGS мікропроцесора 8086/8088.

*Ознака переповнення* OF інформує про те, відбулося переповнення під час виконання операцій чи ні, тобто про правильність результату. Вона дорівнює 0, якщо переповнення не було, в іншому випадку – 1.

*Біт визначення рівня привілеїв уведення/виведення* (I/O P/L) (розряди 12, 13). Це двобітове поле належить до захищеного режиму (про захищений режим див. пп. 2.1.1.4, Б). Рівень привілеїв уведення/виведення (I/O P/L) відображає максимальне значення поточного рівня привілеїв (ПРП). Для максимально допустимого значення ПРП, у разі виконання команд уведення/виведення без генерування спеціального переривання, він також вказує максимальне значення ПРП, яке дає змогу змінити біт IF, якщо нові значення завантажуються зі стеку в регістри FLAGS або EFLAGS. Команди POPF та IRET, якщо вони виконуються при ПРП=0, можуть змінювати поле I/O P/L. Операції вмикання задач завжди змінюють поле I/O P/L, коли нове значення ознаки завантажуються з сегмента стану задачі.

*Ознака вкладки задач* (NT). Цю ознаку використовують у захищеному режимі. NT задають для того, щоб показати, що виконання конкретної задачі вкладено в іншу задачу. Якщо її задано, то сегмент стану поточної вкладки задачі має достовірний обернений зв'язок із сегментом стану попередньої задачі. Цей біт задають або відмінюють командами, що передають керування іншим завданням. Значення NT в EFLAGS перевіряють командою IRET, яка у випадку NT=1 виконує перемикання задач, а у випадку NT=0 це звичайне повернення з переривання.

*Ознаку відновлення* (RF) використовують у покроковому режимі або сумісно з точками переривань регістрів налагодження. Її перевіряють на межі команди перед опрацюванням точки зупинки. Якщо RF=1,

то будь-яка помилка налагодження буде проігнорована на наступній команді. RF автоматично скидається в разі успішного закінчення кожної команди (помилки не зафіксовані), крім команд IRET, POPF, JMP, CALL, INTER, які спричиняють включення задачі. Ці команди задають RF відповідно до збереженої копії EFLAGS у пам'яті.

Наприклад: наприкінці обслуговування програми переривань команда IRET може відновити копію EFLAGS, що має RF=1. Тоді МП відновить виконання програми в адресі точки переривання без генерування іншого переривання в тому ж місці.

*Ознака віртуального режиму VM* забезпечує віртуальний режим МП8086 у межах захищеного. Якщо VM=1, то 32-розрядний процесор перетворюється у високопродуктивний МП 8086. Біт VM може бути заданий тільки в захищеному режимі: командою IRET, якщо поточний привілейований рівень дорівнює 0, або шляхом перемикання задач на будь-якому рівні привілеїв. Біт VM не підпорядкований дії команди POPF, а команда PUSHF завжди встановлює цей розряд в 0, навіть якщо працює у віртуальному 8086 режимі. Копія EFLAGS, збережена у стеку під час опрацювання переривання або під час вмикання задачі, міститиме одиницю в цьому біті, якщо переривання опрацьоване як віртуальне 8086 завдання.

**Сегментні регістри.** Шість 16-розрядних сегментних регістрів містять базові адреси сегментів, що визначають сегменти пам'яті поточного адресування (16-бітові вказівники у реальному режимі та дескриптори – у захищеному). Сегментні регістри показані на рис. 2.8. У захищеному режимі кожен сегмент може мати розміри від одного байта до всього лінійного і фізичного простору машини (до 4Гб). У режимі реального адресування максимальний розмір сегмента обмежений 64 Кб.

15	0	CS						-	-	-
15	0	SS						-	-	-
15	0	DS						-	-	-
15	0	ES						-	-	-
15	0	FS						-	-	-
15	0	GS						-	-	-

Сегментні регістри      Фізична базова адреса (32 біти)      Межа сегмента (32 біти)      Інші атрибути сегмента з описувача

Рис. 2.8. Регістри дескриптора сегмента

Шість сегментів, які адресують пам'ять у будь-який конкретний момент, визначені вмістом регістрів CS, SS, DS, ES, FS і GS. Значення в CS визначає поточний сегмент коду; вміст SS визначає поточний сегмент стеку, а значення в DS, ES, FS, GS – сегменти даних.

У захищеному режимі регістри сегментів містять дескриптори сегментів, які завантажуються автоматично (про дескриптори див. с.92). Регістри дескриптора сегмента невидимі для програміста, однак їхній вміст потрібно знати. Регістр дескриптора співвіднесений з кожним видимим регістром селектора, як показано на рис. 2.8. Кожний з них містить 32-бітову базову адресу сегмента, його межі та інші необхідні ознаки. Коли адреса сегмента завантажується в сегментний регістр, то асоціативний (співвіднесений) регістр дескриптора автоматично модифікується відповідно до нової інформації. У режимі реального адресування модифікується тільки базова адреса шляхом зсування значення сегментного регістра на чотири розряди ліворуч, оскільки максимальна межа й ознаки сегменту фіксовані. У захищеному режимі базова адреса, межа, всі ознаки модифіковані вмістом регістра дескриптора сегмента, індексованого селектором. Кожного разу, коли відбувається посилання на комірку пам'яті, регістр дескриптора сегмента автоматично “втягується” з посиланням на комірку пам'яті; 32-бітова базова адреса сегмента стає компонентом обчислення лінійної адреси, 32-бітове значення межі буде використане для операцій контролю межі, а ознаки – перевірені щодо відповідності типу посилання на комірку пам'яті. До ознак (атрибутів), зокрема, належать: рівень привілеїв, наявність доступу до пам'яті, грануляції, читання чи запис, розмір стека та ін.

**Регістри керування.** МП має три регістри керування довжиною по 32 біти (CR0, CR2 і CR3), які зберігають ознаки процесора і є загальними для всіх задач. Ці регістри, поряд із регістрами системної адреси підтримують стан машини і діють на всі задачі в системі.

31						0
P	Резерв (біти невизначені)				E	T
G					T	S
					E	M
					M	P
					P	E

Рис. 2.9. Регістр контролю CR0

**CR0:** Регістр керування машиною (рис. 2.9). Він містить шість розрядів для керування і визначення стану процесора; 16 молодших розрядів

CR0 також відомі як *Слово Стану Машини* (MSW), яке є присутнє у МП 80286. Для завантаження і зберігання в пам'яті CR0 використовують команди LMSW і SMSW. З метою сумісності з операційними системами МП 80286, команди працюють аналогічно до команд 80286, тобто нові розряди CR0 ігноруються. Нові операційні системи мікропроцесора повинні використати команди для завантаження регістра CR0 повністю: MOV CR0.

Тепер опишемо всі розряди регістра CR0.

*PG (розбиття на сторінки, біт 31).* PG біт задають для того, щоб розблокувати пристрій заміщення сторінок. PG=0, якщо його треба блокувати.

*ET (тип розширення процесора, біт 4).* ET відображає тип розширення процесора (80287 або 80387). За бажанням ET біт може бути скинутий шляхом завантаження CR0 при виконанні програми. Коли ET=1, то застосовується 32-бітовий протокол, інакше – 16-бітовий.

*TS (перемикання задачі, біт 3).* МП задає TS кожного разу, коли виконується операція перемикання задачі та перевіряє його при виконанні команд співпроцесора. Якщо TS=1, то операційний код співпроцесора приводить до операційного переривання “Співпроцесор не готовий” за умови, що біт MP також заданий. Опрацьовувач переривання звичайно зберігає контекст 80287/80387, що належить поточній задачі, і очищає біт TS, перш ніж повернутися до помилкового коду операції співпроцесора.

*EM (емуляція).* Біт EM задають для того, щоб змусити всі коди арифметичних операцій співпроцесора виробляти переривання “Співпроцесор не готовий” (виняток 7). Комбінацію EM=0, MP=1 застосовують у разі наявності співпроцесора, а EM=1, MP=0 – для його програмної емуляції.

*MP (монітор співпроцесора, біт 1).* MP біт використовується разом з TS бітом для визначення того, чи виробляє код операції WAIT помилку “Співпроцесор не готовий” (виняток 7), коли TS=1. Якщо MP=1 і TS=1, то код операції WAIT виробляє виняток 7. В інших випадках цього не відбувається. Зверніть увагу, що TS задано автоматично кожного разу, коли виконується операція перемикання задачі.

*PE (дозвіл захисту, біт 0).* PE біт задають для створення можливості роботи в захищеному режимі (на рівні сегментів). Коли PE=0, то процесор працює в реальному режимі. PE може бути встановлений

шляхом завантаження CR0 або MSW, а скинутий тільки в разі завантаження CR0.

**CR1:** резервний. CR 1 зарезервований для використання в майбутніх моделях процесорів INTEL.

**CR2:** *Лінійна адреса переривання внаслідок відсутності сторінки* (рис. 2.10). CR2 містить 32-бітову адресу, яка визначає наявність помилки на останній сторінці. Код помилки опрацьованої сторінки, поміщений у стек за запитом, може надати додаткову інформацію про статус помилки на сторінці.

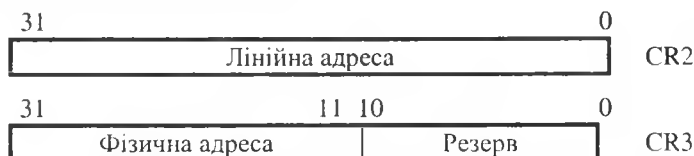


Рис. 2.10. Регістри керування

**CR3:** *Базова адреса таблиці сторінок*. CR3 містить фізичну базову адресу таблиці сторінок. Таблиця сторінок завжди посторінково вирівняна (тобто розташована через 4 Кб). Тому молодші 12 біт CR3 ігноровані і завжди поміщені в пам'ять як невизначені. У процесорі Intel486 два розряди у цьому полі (PWT та PCD) задіяні і відповідають за режим запису і кешування сторінок.

**Регістри системної адреси.** У захищеному режимі є чотири спеціальні регістри для звертання до таблиць або сегмента (рис. 2.11):

GDTR – до таблиці глобального дескриптора GDT;

IDTR – до таблиці дескриптора переривань IDT;

LDTR – до таблиці локального дескриптора LDT;

TR – до сегмента стану задачі TSS.

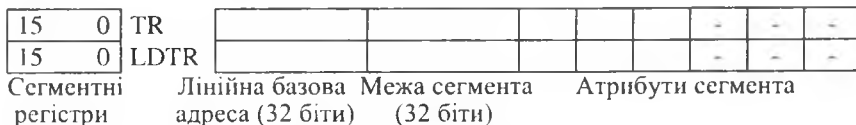
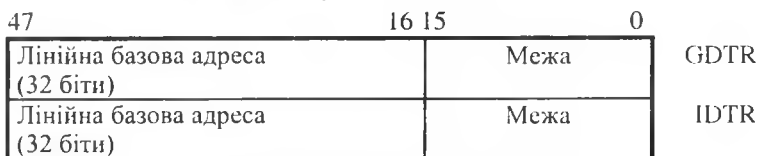


Рис. 2.11. Регістри системної адреси і сегмента

Регістри GDTR і IDTR містять 32-бітову лінійну базову адресу, 16-бітову межу таблиці глобального дескриптора і таблиці дескриптора переривання, відповідно. Сегменти GDT та IDT, оскільки вони глобальні для всіх задач у системі, визначені 32-бітовими лінійними адресами і 16-бітовими значеннями меж.

Регістри LGTR і TR містять 16-бітові адреси сегментів для LDT і TSS. Сегменти LDT і TSS, оскільки вони є сегментами специфічних задач, визначені адресою, що поміщена в регістри системного сегмента.

Зверніть увагу, що регістр дескриптора сегмента (не видимий для програміста) співвідноситься з кожним регістром системного сегмента.

**Регістри налагодження:** Шість доступних для програміста регістрів налагодження забезпечують підтримку процесу налагодження програм. Регістри налагодження DR0-DR3 класифікують чотири лінійні зупинки, тобто містять лінійні адреси заданих контрольних точок. Регістр стану налагодження DR6 використовують для задання точок зупинки, а регістр статусу налагодження DR7 виконує функції керування, а також відображає поточний стан точок зупинки. Використання регістрів налагодження детально описано у [7].

**Тестові регістри.** МП 80386 використовує тільки два регістри для керування і тестування сторінкової переадресації. TR6 – регістр тестування команд, а TR7 – регістр даних, який зберігає результат тестування буфера асоціативної трансляції. Їх використання розглянуто в розділі 2.1.1.8.

**Доступність та сумісність регістрів різних груп.** У доступах до регістрів у реальному і захищеному режимах є певні відмінності (табл. 2.1).

Таблиця 2.1. Доступність регістрів 32-розрядних процесорів

Регістри	Реальний режим		Захищений режим		Режим віртуального 8086	
	Завантаження	Збереження	Завантаження	Збереження	Завантаження	Збереження
Загального призначення	+	+	+	+	+	+
Сегментів	+	+	+	+	+	+
Ознак	+	+	+	+	IO PL	IO PL
Керівні	+	+	PL=0	PL=0	-	+
GDTR, IDTR	+	+	PL=0	+	-	+
LDT, TR	-	-	+	-	-	-
Налагодження	+	+	PL=0	PL=0	-	-
Тестування	+	+	PL=0	PL=0	-	-



Варто пам'ятати про те, що деякі біти регістрів 32-розрядних процесорів невизначені. Це необхідно для сумісності програмного забезпечення, написаного для молодших моделей процесорів, з програмним забезпеченням наступних процесорів. У разі викликання невизначених бітів треба дотримуватися таких рекомендацій.

1. У разі тестування значень бітів регістрів уникати залежності від стану невизначених бітів. Ліпше їх замаскувати.

2. Уникати залежності від стану будь-якого із невизначених бітів у випадку розміщення їх у пам'яті або в іншому регістрі.

3. Уникати залежності від можливості розміщувати інформацію, записану в будь-якому з невизначених розрядів.

4. Під час завантаження регістрів завжди завантажувати їх як нулі.

5. Регістри, поміщені раніше, можуть бути перезавантажені без маскування.

### 2.1.1.3. Переривання роботи мікропроцесора

*Переривання* – це тимчасове припинення виконання поточної програми, яке відбувається апаратно. Після цього повинна настати реакція на подію, тобто опрацювання переривання.

Використовують переривання з такою метою:

1) для збільшення ефективності роботи центрального процесора;

2) для доступу до апаратних засобів і програм, які контролює операційна система;

3) для виконання операцій введення/виведення незалежно від процесора (унаслідок різної швидкості виконання).

Розрізняють внутрішні і зовнішні переривання, залежно від природи їх появи, та масковані і немасковані залежно від способу реагування на їхню появу.

*Маска переривання* – засіб вибіркового подавлення переривання у тих випадках, коли їх опрацювання можна відкласти на пізніший час. Ось деякі асоціації з побутовими подіями.

*Приклад 1.* Ви ведете телефонну розмову і несподівано хтось дзвонить у двері. Ви можете проігнорувати цей дзвінок і продовжувати розмову далі; можете негайно перервати розмову і піти вняснити, хто дзвонить у двері, а потім продовжити розмову; можете продовжити розмову до того часу, поки не знайдете зручний момент, для того щоб її перервати, а потім піти до дверей. Усі ці приклади асоціюються з поняттям

маскованих переривань у мікропроцесорній системі, якщо уявити собі, що телефонна розмова – це процес виконання програми, дзвінок у двері – зовнішнє переривання.

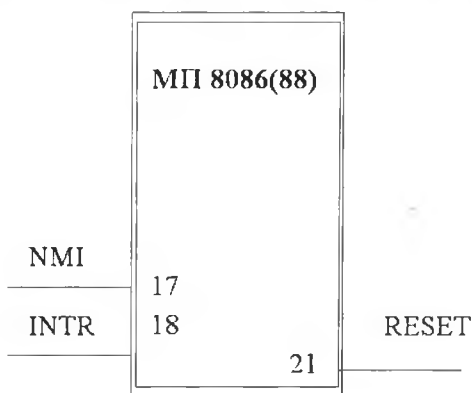
*Приклад 2.* Оператор веде спостереження за функціонуванням складного об'єкта і несподівано фіксує сигнал пожежної тривоги. Оператор припиняє спостереження, негайно викликає пожежну службу і тільки після цього повертається до спостереження за об'єктом. Це аналогія немаскованих переривань у процесорі.

Немасковані переривання МП мають найвищий пріоритет, а мікропроцесор негайно обслуговує їх. Такими перериваннями є, наприклад, у монопольному режимі персонального комп'ютера – помилка пам'яті чи збій живлення; у колективному – перевищення відведеного поля пам'яті, сигнал системного годинника на завершення роботи тощо.

Процесор 8086(88) автоматично перериває роботу в таких випадках (*внутрішні переривання*):

- у разі ділення на нуль;
- під час виконання програми, якщо ознака  $TF = 1$  (покроковий режим);
- у разі виконання команди з шістнадцятковим кодом  $CC$ . Це так зване переривання у заданій точці програми, що виникає під час налагодження програми.

Зовнішні переривання – це переривання від пристроїв введення/виведення, наприклад, клавіатури. Є три лінії переривань: RESET, NMI і INTR:



У разі появи сигналу в лінії RESET центральний процесор виконує такі дії:

- встановлює ознаку  $IF = 0$ . Це унеможливорює виконання маскованих і покровових переривань;
- зануляє вказівник команд  $IP$ ;
- зануляє регістри сегментів  $DS, ES$  і  $SS$ ;
- засилає шістнадцятковий код  $FFFF$  у регістр сегмента команд.

Центральний процесор починає роботу з комірки  $FFFF0$ , де міститься команда  $JMP$ , яка передає керування процедурі ініціалізації, що і запускає в роботу комп'ютер.

Для ідентифікації процедур, необхідних для обслуговування вимог переривання, слугують **вектори переривань**. Кожний зовнішній вимозі може бути поставлений у відповідність код від 0 до 255. Таблиця векторів переривань (табл. 2.2) займає 1024 молодші байти пам'яті, тобто комірки з фізичними адресами пам'яті від 0 до  $03FFh$ . Кожний вхід таблиці є вказівником подвійного слова, яке містить початкову адресу процедури для обслуговування переривання цього типу. Старше 16-бітове слово містить базову адресу сегмента, а молодше — адресу процедури всередині сегмента.

**Таблиця 2.2. Таблиця векторів переривань**

Адреса пам'яті	16-розрядний вказівник	Номер вектора переривання
03FCh	вказівник типу 255	255
03F8h	вказівник типу 254	254
.....	.....	.....
0010h	вказівник типу 4	4 (переповнення)
000Ch	вказівник типу 3	3 (один байт містить CCh)
0008h	вказівник типу 2	2 (немасковане переривання)
0004h	вказівник типу 1	1 (покрововий режим)
0000h	вказівник типу 0	0 (помилка ділення)

*Масковані переривання* — це зовнішні переривання, що надходять по лінії  $INTR$ . Команда завжди виконується до кінця і тільки після цього починається опрацювання запиту на переривання.

Якщо ознака переривання  $IF=0$  (переривання неможливе), то МП ігнорує запит. Якщо ознака дорівнює 1 (переривання дозволене), то МП підтверджує вимогу переривання і передає керування тій процедурі, яка повинна обслужити вимогу.

Для цього МП виконує вісім таких операцій:

1) генерує сигнал підтвердження зовнішнього переривання. Цей сигнал повідомляє зовнішній пристрій, що його вимога прийнята;

2) зчитує код переривання, який надходить на інформаційну шину від зовнішнього пристрою;

3) вміст регістра ознак записує в комірку пам'яті, адреса якої зберігається у поточній вершині стеку, яка визначається регістрами SS і SP;

4) занулює ознаку переривань (це запобігає появі нового переривання);

5) занулює ознаку пастки TF (це робить неможливим покроковий режим);

6) у стек завантажується вміст вказівника команд IP;

7) у вказівник команд IP засилається 16-бітове слово, яке міститься за такою адресою (фізичною):

(тип переривання \*4) і (тип переривання \*4)+1;

8) у регістр сегмента команд CS засилається 16-бітове слово, яке міститься у комірках пам'яті з фізичними адресами:

(тип переривання \*4)+2 і (тип переривання \*4)+3.

Після виконання двох останніх операцій керування передається процедура обслуговування переривань, яка містить машинні команди, потрібні для задоволення маскованого переривання.

*Немасковані переривання.* Ці переривання надходять у МП по лінії NMI (від'єднання живлення, збій пам'яті та ін.). МП у відповідь на сигнал NMI виконує таку послідовність операцій:

- вміст регістра ознак записує в комірку пам'яті, адреса якої зберігається у поточній вершині стеку, яка визначається регістрами SS і SP;

- занулює ознаку переривань (це запобігає появі нового переривання);

- занулює ознаку пастки TF (це робить неможливим покроковий режим);

- у стек завантажується вміст регістра сегмента команд CS;

- у стек завантажується вміст вказівника команд IP;

- 16-бітове слово з комірки з фізичною адресою 00008h записує в регістр IP;

- 16-бітове слово з комірки з фізичною адресою 0000Ah записує в регістр CS.

Після цього, як і для маскованих переривань, виконується процедура обслуговування.

### Типи переривань і винятків 32-розрядних МП

Як зазначено, переривання, спричинені машинним обладнанням, виникають унаслідок зовнішніх подій і бувають двох типів: масковані і немасковані. Вони опрацьовуються після виконання поточної команди. Після того, як програма опрацювання переривань закінчує обслуговувати переривання, виконання програми продовжується з команди, яка є наступною. Відмінності між маскованими і немаскованими перериваннями описано нижче.

**Винятки** класифікують як помилки (або відмови), пастки або (передчасні) припинення залежно від способу їхнього повідомлення і від того, чи підтримується повторний запуск команди, яка викликає виняток.

*Помилки (fault)* – це винятки, які виявляються і виконуються до виконання команди з помилкою. Наприклад, *помилка* відбудеться в системі віртуальної пам'яті, коли процесор зробить посилання на сторінку або сегмент, яких не існує. Операційна система вибере сторінку або сегмент з диска, після чого процесор перезапустить команду.

*Пастки (trap)* – винятки, повідомлення про які надходять негайно після виконання тієї програми, яка викликає виняток. До класу пасток належать і програмні переривання.

*Припинення, або аварійні завершення (abort)* – винятки, які не дозволяють визначати точно місце знаходження команди, яка викликає виняток. Їх використовують для повідомлення про грубі помилки, такі як апаратна помилка або неправильні значення в системних таблицях.

Отже, коли програма обслуговування переривання закінчує роботу, то виконання продовжується з команди, яка є наступною після команди, що спричинила переривання. З іншого боку, адреса повернення з програми помилки винятку завжди вкаже на команду, яка викликає виняток, і буде містити будь-який з префіксів провідної програми.

У табл. 2.3 наведені всі можливі переривання для 80386 і відображено, куди відсилає адреса повернення. Процесор здатний опрацювати до 256 різних переривань/винятків. Для обслуговування переривань необхідно означити (скласти) таблицю з 256 векторами переривань. **Вектори переривань** – це вказівники на відповідні програми обслуговування; у реальному режимі вектори – це 4-байтові величини: сегмент коду плюс 16 бітів зміщення; у захищеному режимі вектори переривань є 8-байтовими величинами, які поміщені в таблицю дескрипторів

переривань. Із 256 можливих переривань 32 зарезервовані для використання INTEL, решта 224 вільні для застосування користувачем.

Багато винятків, що відображені в таблиці переривань і розглянуті у першій частині, не застосовують у реальному режимі. Зокрема переривання 10, 11, 12, 14 не можуть відбутися в реальному режимі. Інші переривання у реальному режимі мають деякі відмінності.

Таблиця 2.3. Вектори переривань

Функція	Номер переривання	Команда, яка може викликати виняток	Адреса означає повернення на помилкову команду	Тип
Помилка ділення	0	DIV, IDIV	Так	Помилка
Виняток налагодження	1	Будь-яка	Так	Пастка
NMI переривання	2	INT2 або NMI	Ні	NMI
Однобайтове переривання	3	INT3	Ні	Пастка
Переповнення	4	INT0	Ні	Пастка
Контроль межі масиву	5	Межа	Так	Пастка
Неправильний код операції	6	Будь-яка заборонена команда	Так	Пастка
Відсутність пристрою	7	ESC, WAIT	Так	Пастка
Подвійна помилка	8	Будь-яка команда, здатна виробити виняток	Так	Припинення
Перевантаження сегмента співпроцесора	9			Пастка
Неправильний TSS	10	JMP, CALL, IRET, INT	Так	Помилка
Відсутній сегмент	11	Команди з сегментними регістрами	Так	Помилка
Помилка стека	12	Посилання на стек	Так	Помилка
Загальна помилка захисту	13	Будь-яке звертання до пам'яті	Так	Помилка
Перерив. внаслідок відсутності сторінки	14	Будь-який доступ до пам'яті для вибору коду	Так	Помилка
Помилка співпроцесора	16	ESC, WAIT	Так	Помилка
Резерв	18-32			
Програмні переривання	0-225	INTn	Так	Пастка

### Опрацювання переривань

Алгоритм опрацювання переривань процесора описаний вище. Згадаємо, що адреса поточної команди і регістр ознак EFLAGS зберігаються в стеку, що дає змогу відновити перервану програму. Процедура опрацювання переривання визначена 8-бітовим вектором таблиці переривань, який містить початкову адресу програми опрацювання переривань. Після цього виконується викликана програма опрацювання. Старий стан процесора відновлюється командою IRET і за адресою повернення (тобто за адресою, збереженою у стеку) відновлюється виконання програми. Восьмибітовий вектор переривання подається на процесор кількома способами:

- винятки генерують і передають вектор переривання внутрішньо;
- команди INT містять або включають у себе вектор;
- переривання, масковані обладнанням, “транспортують” 8-бітовий вектор через шину підтвердження переривань від зовнішнього контролера.

### Масковані переривання

Масковані переривання – найбільш загальний засіб, який використовує процесор для відповіді на асинхронні зовнішні події в обладнанні. Апаратне переривання відбувається тоді, коли біт IF ознаки переривань розблокований (дорівнює 1), і рівень INTR на виході процесора є високим. Процесор реагує на масковані переривання тільки між виконанням команд. Рядкові команди мають “вікна переривань” між переміщеннями пам’яті, що робить можливим переривання під час опрацювання довгих рядків. Коли виникає переривання, процесор зчитує 8-бітовий вектор, що надходить від обладнання, який визначає джерело переривання (одне із 224 визначених користувачем переривань).

Коли відбувається обслуговування переривання, то біт IF у регістрі EFLAGS скинутий. Це дає змогу ефективно запобігати додатковим перериванням під час опрацювання обслуговування переривань. Однак IF може бути заданий опрацьовувачем переривань, щоб дозволити вкладення переривань. Коли команда IRET виконана, відновлюється початковий стан ознак, а отже, і біта IF.

### Немасковані переривання

Немасковані переривання забезпечують обслуговування переривань дуже високого рівня. Як і для попередніх процесорів, одним із загальних прикладів немаскованих (NMI) переривань може слугувати переривання

за збоєм живлення. Коли активізований контакт NMI, то відбувається переривання згідно з вектором 2. На відміну від звичайних переривань, для NMI не виконується послідовність підтверджень приймання переривань.

Під час процедури обслуговування NMI процесор не обслуговуватиме ні подальший запит NMI, ні запити INT доти, доки не буде виконана команда повернення із переривання (IRET) або поки процесор не буде скинутий. Якщо NMI відбудеться під час обслуговування NMI, то його (переривання) присутність буде збережена для опрацювання після опрацювання поточного NMI, тобто після першої ж команди IRET. Біт IF очищається на початку NMI для блокування подальших INTR команд.

### Програмні переривання

Третім типом переривань/винятків є програмні переривання. Команда INT  $n$  змушує процесор виконувати програму обслуговування, яку визначає вектор  $n$  у таблиці переривань.

Особливим випадком двобайтового програмного переривання INT  $z$  номером  $n$  є однобайтове переривання INT3 або переривання зупинки. Шляхом введення цієї однобайтової команди в програму користувач має змогу задати точки зупинки в своїй програмі для її налагодження.

Ще одним типом програмного переривання є переривання покрокового режиму. Воно корисне для відлагоджування програм.

### Пріоритетність переривань

Оскільки переривання розпізнають тільки на межах команд (тобто тоді, коли одна команда закінчується, а інша починається), то можливо, що одночасно активними можуть бути кілька переривань. У випадку одночасних переривань вони опрацьовуватимуться згідно з таким пріоритетом:

Пріоритет опрацювання	Переривання/виняток
1 (вищий)	Помилка винятку
2	Команда пастки
3	Пастка налагодження для заданої команди
4	Помилка налагодження для наступної команди
5	MNI переривання
6	INTR переривання

*Приклад:* задана команда викликає системне переривання налагоджень і виняток “сегмент відсутній”. Процесор насамперед відреагує на виняток “сегмент відсутній” (11), спробувавши активізувати опрацьовувач винятку 11. Програма опрацювання винятку 11 буде перервана, внаслідок чого адреса опрацьовувача винятку 11 збережеться в стеку.



Після цього буде викликаний опрацьовувач налагодження і керування знову повернеться до опрацьовувача винятку 11. Це дає змогу системо-техніку відлагоджувати свої опрацьовувачі винятків.

#### 2.1.1.4. Організація пам'яті та адресування

##### а) 16-розрядні процесори

Як зазначено, Intel8086(88) має пам'ять 1 Мб. Для того, щоб виконати адресування для такої кількості комірок, необхідно мати 20-розрядний вказівник. Оскільки вказівник команд IP (як і інші регістри МП) має 16 розрядів, то МП *не має* прямого доступу відразу до всієї пам'яті. Це завдання вирішують за допомогою **сегментування пам'яті**, тобто поділу її на частини.

##### Організація сегмента

Поділ виконано так, що кожний сегмент містить до 64 Кб (65 536 байт). Початкова адреса може бути задана прикладною програмою і завжди повинна починатися з 16-байтових меж. Нагадаємо, що обмін інформацією між МП і пам'яттю відбувається за допомогою фізичних адрес.

Програма ж частіше використовує логічні адреси, що дає змогу записувати команди без попереднього визначення місця, де ця команда буде розташована у пам'яті. Логічна адреса має дві складові: *базу сегмента* і значення *зміщення* всередині сегмента. Як тільки блок спраження звертається до пам'яті, суматор адреси (СМА) сформує фізичну адресу за правилом: *значення бази сегмента зміщується на чотири розряди ліворуч і отримане 20-розрядне число додається до зміщення*.

Базові адреси сегмента отримують діленням фізичної адреси початкової комірки сегмента на 16. Ці адреси містяться у сегментних регістрах (отже, МП 8086 може одночасно обслуговувати до чотирьох сегментів). На розташування сегмента накладається тільки одне обмеження: він повинен бути на межі шістнадцяти байт (тобто фізична адреса початкової комірки повинна ділитися на 16).

Якщо сегмент певним чином визначений, то регістри-вказівники можуть бути використані для адресного звертання до будь-якого байта чи слова в межах цього сегмента. Отже, МП має доступ тільки до 64 Кб пам'яті, а адресування до 1 Мб можливе тільки із застосуванням регістрів сегментів. Це один із недоліків МП сім'ї 8086 (88). Наприклад, МП Motorola MC 68000 (містить 32-розрядні регістри-вказівники) може безпосередньо адресувати до 16 Мб пам'яті.

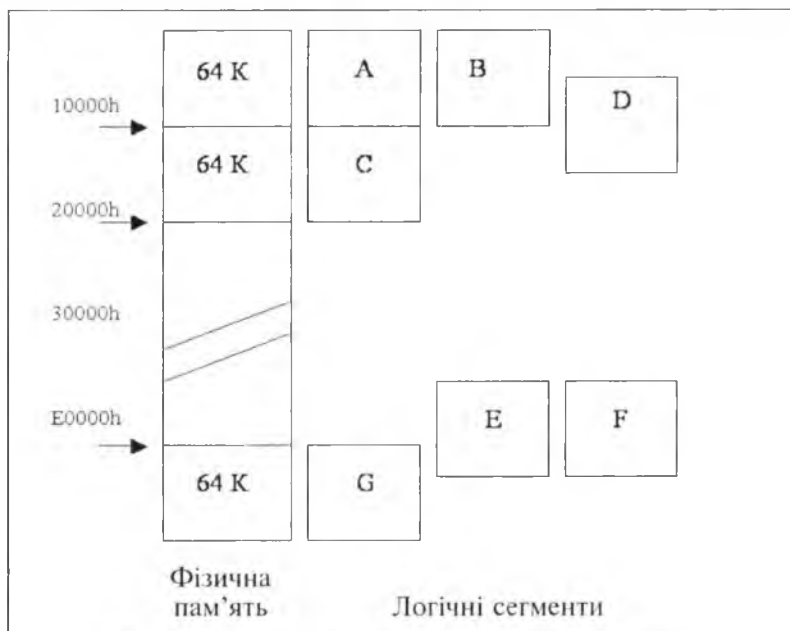


Рис. 2.12. Приклад розташування сегментів пам'яті.

Як зазначено вище, МП має чотири регістри сегментів. Отже, одночасно виділяють чотири адресовані сегменти :

- командний CS;
- даних DS;
- додатковий ES;
- стековий SS.

Сегменти можуть бути суміжними (А, С), розділеними (С, Е), перекриватися частково (В, D; Е, G) або повністю (А, В чи Е, F).

Командний сегмент містить машинні команди, які вибирає пристрій зв'язку, а виконує виконавчий пристрій.

Сегмент даних використовують для запам'ятовування програм, змінних і сталих.

Додатковий сегмент забезпечує додаткові 64 Кб робочої пам'яті.

Стековий сегмент містить стекову структуру ємністю до 64 Кб.

Якщо у всі регістри сегментів завантажити одну й ту ж базову адресу, то всі чотири сегменти відображатимуть одну й ту ж область фізичної пам'яті.

Сегментація ефективна для модульної побудови програмного забезпечення і структурного програмування, коли великі програми розбиваються на дрібні модулі.

### Режими адресування

Пам'ять процесора можна уявити собі як послідовність байтів, наймолодший з яких має адресу 00000h, а найстарший – FFFFFh. Для адресування слова і подвійного слова діють такі домовленості:

1) 16-бітове слово вводиться в пам'ять так, що старший байт міститься у комірці з більшим номером;

2) 32-бітове подвійне слово вводиться складніше, а власне – з використанням його як вказівника для адресування байта чи слова у будь-якому сегменті. Молодше слово адреси означає адресу переміщення, а старше – початкову адресу сегмента. Два байти кожного слова вводяться за правилом запису 16-розрядного слова.

Процесор має досить гнучку систему адресування, що дає змогу ефективно використовувати різні форми задання операндів у командах програми. Система адресування забезпечує використання регістрових та безпосередніх операндів, пряме та непряме адресування, базово-індексне адресування зі зміщенням і без нього, адресування рядків даних, адресування портів введення/виведення.

*Регістрові операнди* – це приклад найефективнішого виконання команд процесора, тому що операції з операндами відбуваються без звертання до пам'яті. Джерелом, приймачем, або і тим, і іншим можуть бути регістри загального призначення. Сегментні регістри можуть бути або тільки джерелом, або тільки приймачем операндів.

Наприклад, команда (у дужках записано машинний код команди)

#### **AND AL, DL (20D0)**

виконає операцію логічного множення вмісту регістрів AL і DL, а результат помістить у регістр AL. Процесор виконає її за три такти.

*Безпосередні операнди* – визначають як частину самої команди. Дані можуть бути 8- або 16-бітовими. Приймачем найліпше слугує регістр AL (AX). Наприклад, нам потрібно порівняти вміст регістра AL зі сталою 01. Машинна команда у цьому випадку має вигляд

#### **CMP AL, 01 (3C01),**

і процесор виконає її за чотири такти.

Перш ніж перейти до ознайомлення з іншими режимами адресування пам'яті процесора, з'ясуємо поняття виконавчої адреси.

**Виконавчою адресою** (ВА) операнда називають адресу всередині сегмента, яку процесор обчислює за певними правилами. ВА є переміщуваним 16-розрядним числом без знака, що забезпечує доступ до кожного байта сегмента. Щоб зрозуміти процес формування виконавчої адреси, необхідно розглянути структуру команди процесора:

Код операції	Mod	Reg	R/M	Зміщення	Безпосередній операнд
	XX	YYY	ZZZ		

Перед полем коду операції можлива наявність префіксного поля. Поле Mod (від *Mode* – режим) потрібне для того, щоб відрізнити, чи відповідають операнди команд регістрам ЦП або коміткам пам'яті. Величина XX має чотири можливі двійкові значення: 00, 01, 10, 11. Двійкове значення 11 у полі Mod свідчить про те, що операндами машинних команд є регістрові операнди. Двійкові значення 00, 01 і 10 задають типи адресації відповідно з нульовим, одно- або двобайтовим зміщенням. Зміщення може бути 8-бітовим числом зі знаком або 16-бітовим числом без знака, яке входить у машинну команду як її частина і яке використовують у разі обчислення виконавчої адреси.

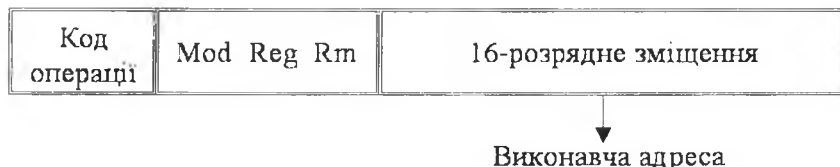
Позначення YYY відповідає трьом бітам, які утворюють поле Reg (від *Register* – регістр). Його застосовують для визначення регістра, у якому міститься операнд. Поле Reg ідентифікує 8-бітовий регістр під час роботи зі словами. У деяких командах ці біти використовують також для ідентифікації групи, до якої належить машинна команда (наприклад, команда AND у групі логічних команд).

Три біти ZZZ формують поле Rm (від *Register/Memory* – регістр/пам'ять). Разом із полем Mod його застосовують для ідентифікації певного типу адресації. У команді допустима будь-яка комбінація в полях Mod і Rm, що забезпечує різноманітність режимів адресації пам'яті в сім'ї мікропроцесорів 8086/8088.

**Пряме адресування** є найпростішим типом адресації – воно не використовує ніяких регістрів. Виконавчу адресу беруть безпосередньо з 16-бітового поля зміщення машинної команди. Ця пряма адреса однозначно визначає байт або слово пам'яті, які розміщені всередині сегмента.

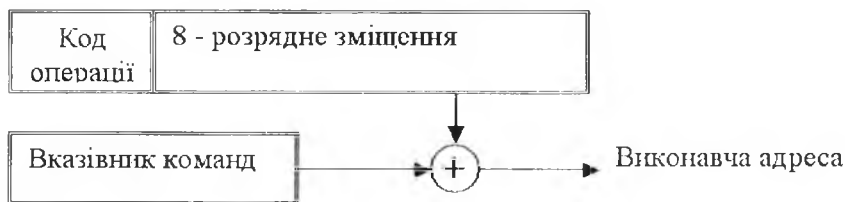
Пряму адресацію переважно використовують для роботи з простими змінними або сталими. У процесорі 8086(88) повинна виконуватись умова, згідно з якою пряма адреса не виходить за межі сегмента даних. Оскільки в програмах пряму адресу можна визначати стосовно

будь-якого з чотирьох доступних сегментів, то для його модифікації можливе застосування префіксних команд. У разі використання прямої адресації в полі Mod міститься число 00, а в полі Rm – число 110.



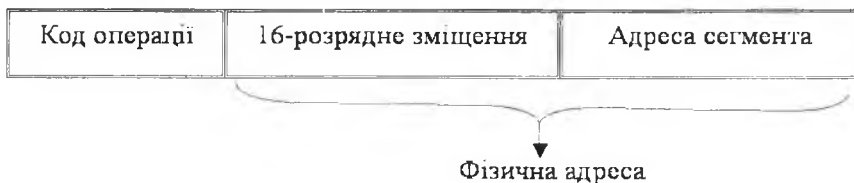
Для прямої адресації використовують два спеціальних види адресації: відносну та абсолютну.

У разі відносної адресації поле зміщення відображене 8-бітовим числом зі знаком. Виконавча адреса в цьому разі визначена сумуванням вмісту поля зміщення і регістра вказівника команд IP:



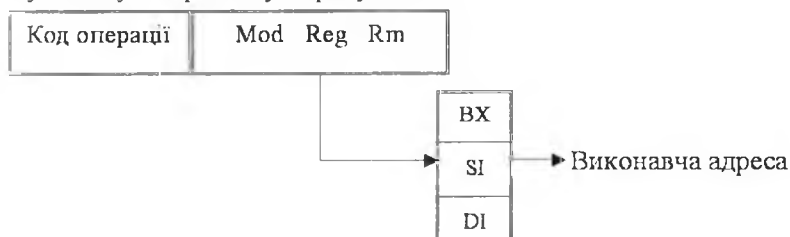
Відносну адресацію використовують для команд умовних переходів, таких як JE (перехід по рівності), JO (перехід по переповненню). У разі використання відносної адресації байт адресації (Mod/Reg/Rm) не потрібний.

У разі абсолютної адресації частина команди є 32-бітовим вказівником, який визначає фізичну адресу в пам'яті МП 8086(88). Молодше слово вказівника трактується в цьому випадку як переміщувана адреса сегмента, базовою адресою якого є старше слово вказівника:



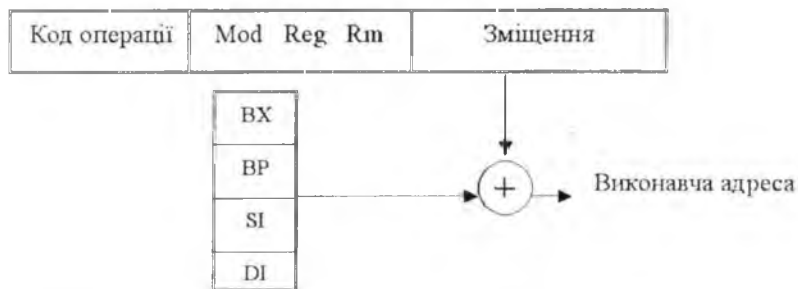
Для абсолютної адресації наявність байта адресації (Mod/Reg/Rm) також не потрібна.

*Непряме адресування.* За такої адресації виконавчою адресою є вміст базового або індексних регістрів (BX, SI або DI). Завдяки цьому одна і та ж команда може звертатися до кількох різних ділянок пам'яті через просту зміну вмісту базового або індексного регістрів, які беруть участь у непрямому адресуванні.



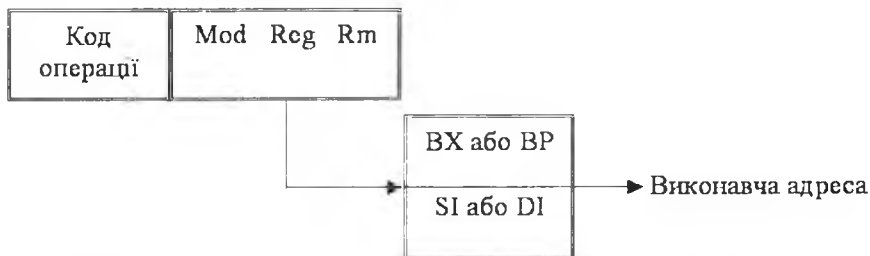
У разі непрямого адресування є один виняток: для команд JMP (перехід) і CALL (виклик процедур) виконавчою адресою може бути вміст будь-якого з 16-бітових регістрів загального призначення (AX, BX, CX, DX, SI, DI, BP, SP).

*Непряме адресування зі зміщенням.* У разі такої адресації ВА визначена як сума зміщення і вмісту одного з регістрів – базового або індексного (BX, BP, SI, DI). Зміщення може бути 8-бітовим числом зі знаком або 16-бітовим без знака. У разі використання регістрів BX, SI за виконавчу адресу приймають переміщувану адресу поточного сегмента даних, базова адреса якої визначена регістром сегмента даних DS. Якщо ж використовують регістр BP, то за виконавчу адресу приймають переміщувану адресу поточного стекового сегмента, базова адреса якого міститься в регістрі сегмента стеку SS:



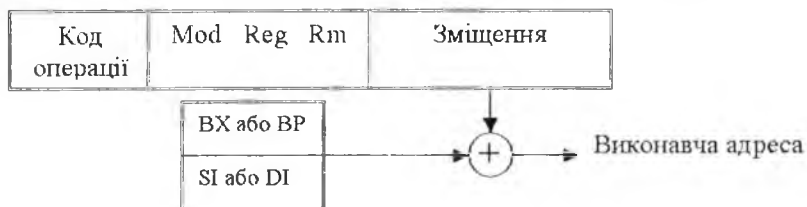
Оскільки в прикладних програмах можуть бути використані префіксні команди, то є можливість переприсвоєння виконавчих адрес для інших сегментів пам'яті.

*Базово-індексне адресування.* За такого адресування виконавча адреса визначена як сума вмісту базового (BX або BP) та індексного регістрів (SI або DI). Завдяки можливості зміни під час виконання програми вмісту базового та індексного регістрів базово-індексне адресування є дуже гнучким засобом доступу до різних ділянок комп'ютерної пам'яті. У разі використання регістра BX виконавча адреса визначена щодо сегмента даних з базовою адресою, яка міститься в регістрі DS, а у разі застосування регістра BP – щодо сегмента стека, базова адреса якого міститься в регістрі SS.



У разі базово-індексного адресування поле Mod визначене числом 00, поле Rm – двійковим числом 000, 001, 010 чи 011, відповідно до того, яке адресування використано: [BX+SI], [BX+DI], [BP+SI] чи [BP+DI].

*Базово-індексне адресування зі зміщенням.* У разі такого адресування виконавча адреса визначена як сума вмісту базового регістра (BX чи BP), індексного регістра (SI чи DI) і розміру зміщення. Як зазначено, розмір зміщення є частиною команди і може бути 8-бітовим числом зі знаком чи 16-бітовим без знака. За виконавчу адресу приймаємо переміщувану адресу поточного сегмента даних чи стекового сегмента залежно від того, який з регістрів використовуємо – BX чи BP:

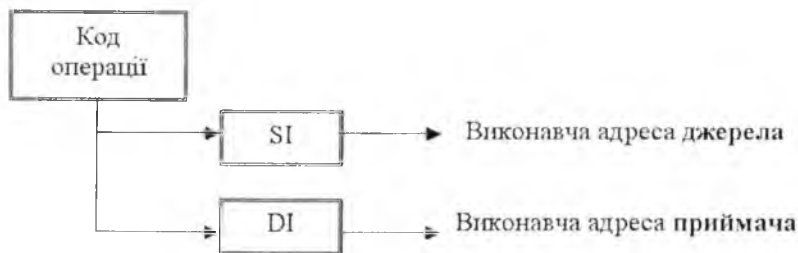


Прикладні програми можуть мати префіксні команди, завдяки яким можливо переприсвоювати виконавчі адреси для інших сегментів пам'яті.

У разі базово-індексного адресування зі зміщенням поле Mod містить двійкове число 01 чи 10 залежно від того, як визначено зміщення: 8- чи 16-бітовим числом. Поле Rm таке саме, як і в разі базово-індексного адресування.

*Адресування рядків даних.* Рядкові команди, такі як MOVS (переслати рядок), CMPS (порівняти рядок), LODS (завантажити рядок), STOS (записати рядок) і SCAS (сканувати рядок) не використовують жодного з розглянутих типів адресації для вибору своїх операндів. Вміст індексних реєстрів (SI і/або DI) використовують для безпосереднього визначення потрібної ділянки пам'яті. Регістр SI завжди використовують як вказівник першого байта чи слова рядка-джерела.

Регістр DI застосовують як вказівник першого байта чи слова рядка-отримувача. Команда LODS пропонує використання реєстра SI як вказівника джерела. Команди STOS і SCAS використовують реєстр DI як вказівник отримувача. Команди MOVS і CMPS застосовують обидва реєстри. Переміщувана адреса, яка є в реєстрі DI, завжди стосується поточного додаткового сегмента. Для команд робіт з рядками даних можна визначити і префіксні команди, які дають змогу переприсвоювати сегмент даних (для команд LODS, MOVS і CMPS) чи додатковий сегмент (для команди SCAS) іншому сегменту, який міститься в будь-якій частині пам'яті 8086(88):



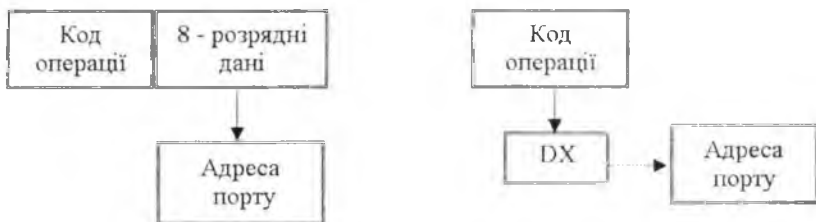
Під час виконання рядкової команди збільшення або зменшення вмісту реєстрів SI чи DI визначене станом (нульовий чи одиничний) ознаки напрямку. Залежно від того, з чим працює команда – з байтом чи зі словом, – вміст індексних реєстрів збільшується/зменшується, відповідно, на 1 чи 2.

*Адресування портів введення/виведення.* Цей тип адресування забезпечує доступ до пристроїв введення/виведення (У/В), які приєднані до У/В області пам'яті 8086(88). Область У/В, як доповнення до



мегабайтової пам'яті, містить ще 65 Кб. Ця область не є сегментованою і її можна використовувати як окремо, так і разом з усією пам'яттю 8086(88). В останньому випадку зменшується мегабайтова пам'ять на 64К, однак забезпечена додаткова гнучкість під час програмування, оскільки всі розглянуті раніше типи адресування можуть забезпечити доступ до цієї області.

Кожний байт області У/В 8086(88) може бути призначенням як адреса однозначно визначеного порта У/В. Отже область У/В може обслуговувати 64 К (65536) 8-бітових портів У/В. Дані можуть бути передані між накопичувачем і будь-яким портом У/В, який адресується всередині 64-кілобайтної області У/В. Регістр AL використовують для передавання байта, а регістр AX – для передавання слова. Для адресування порту У/В є команди IN (ввести з порту) і OUT (вивести в порт). Якщо номер порту перебуває в межах 0-255 (десяткових чисел), то його можна визначити частиною машинної команди; якщо номер порту перевищує 255, то для його зберігання використовують регістр DX:



*Приклад:* нехай порт з адресою 3D5h є портом введення в буферну пам'ять пристрою друку. Виведення у буфер коду ASCII відбуватиметься за допомогою такої послідовності команд:

MOV DX, 3D5h	завантажити адресу порту у DX
MOV AL, 1Bh	код 1B у регістр AL
OUT AL, DX	вивести код у буферну пам'ять.

**Стекова пам'ять.** **Стек** – це область пам'яті, спеціально виділена для тимчасового зберігання параметрів чи програмних даних, які необхідні для інформаційного зв'язку програм і процедур. Запис і читання даних у стеку організовані за принципом “першим прийшов – останнім пішов”. У стек може бути завантажено в певній послідовності низку даних, які потім вибирають (зчитують) зі стеку, проте вже в зворотній послідовності. Кількість стеків у пам'яті ЕОМ обмежена виділеною областю пам'яті. Максимальна ємність стекової структури становить

32К (32768) 16-бітових слів. Отже стек може повністю розміститися в одному сегменті пам'яті.

Оскільки є тільки один регістр сегмента стеку **SS**, який зберігає початкову (базову) адресу стеку, то в будь-який момент часу можна звертатися тільки до одного стеку. Вказівник стеку **SP** використовують для зберігання адреси останнього члена послідовності (вершини стеку), який записано в стек.

#### **б) 32-розрядні процесори. Захищений режим.**

##### **Механізми адресування**

Усі можливості процесора розкриваються у захищеному режимі віртуальної адреси (захищений режим). Цей режим значно збільшує адресний простір (до 4 Гб) і дає змогу виконання програм практично не обмеженого розміру (64 Тб). Крім того, він дає змогу працювати з усіма 8086 і 80826 програмами, забезпечуючи керування величезною пам'яттю та апаратним механізмом захисту. Захищений режим забезпечує використання додаткових команд, спеціально оптимізованих для захищених багатозадачних операційних систем. Базова архітектура 32-розрядного процесора незмінна; регістри, команди і режими адресування, що описані в попередніх розділах, збережені.

Головна відмінність між захищеним і реальним режимом полягає, з погляду програміста, в тому, що збільшується адресний простір і змінюється механізм адресування.

Пам'ять 32-розрядних процесорів розділена на величини: 8-бітові (байти), 16-бітові (слова) і 32-бітові (подвійні слова). Слова зберігаються в двох послідовних байтах пам'яті так, що байт молодшого порядку розміщений у молодших адресах, а старший – у старших. Подвійні слова зберігаються в чотирьох послідовних байтах пам'яті з байтом молодшого порядку в наймолодшій адресі, а байт старшого порядку – у найстаршій. Адресою слова або подвійного слова є адреса байта молодшого порядку.

Крім цих головних типів даних, 32-розрядний процесор підтримує два великі блоки пам'яті: сегменти і сторінки. Пам'ять може бути розбита на один або декілька сегментів різної довжини, які можуть бути перенесені на диск або розподілені між програмами, або ж організована в одну чи більше сторінок (розмір сторінки має бути кратний  $2^n$  Кб). Нарешті, сегментація і розбиття пам'яті на сторінки можуть бути об'єднані, що дає змогу використати переваги обидвох систем. Сегментація

корисна в разі організації пам'яті в логічні модулі, і є інструментом для програміста прикладних програм, тоді як сторінки використовують на системному рівні для керування фізичною пам'яттю.

Як і в реальному, в захищеному режимі використовують дві компоненти для формування логічної адреси: 16-бітовий селектор для визначення лінійної базової адреси сегмента, і 32-бітова виконавча адреса, яку додають до базової адреси для формування 32-бітової лінійної адреси. Після того лінійну адресу використовують або як 32-бітову адресу або, якщо є можливість сторінкової організації пам'яті, механізм підкачування перетворює 32-бітову лінійну адресу в 32-бітову фізичну.

Спосіб обчислення базової адреси у кожному режимі свій. У захищеному режимі селектор використовують для специфікації індексу в таблиці операційної системи. Таблиця містить 32-бітову базову адресу цього сегмента. Фізична адреса формується шляхом додавання базової адреси, отриманої із таблиці, до зміщення.

Підкачування (див. стор. 98) забезпечує додатковий механізм керування пам'яттю, який працює тільки в захищеному режимі, а також керування дуже великим сегментом процесора. Механізм підкачування переводить захищену лінійну адресу, яку видає блок сегментування, у фізичну.

### Поняття про адресні простори

32-розрядний процесор має три різні адресні простори: логічний, лінійний і фізичний. Логічна адреса (її також називають віртуальною адресою) складається із селектора сегмента і зміщення. *Селектор* – це вміст сегментного регістра. Зміщення формується шляхом сумування всіх адресних компонентів (Бази, Індекса, Зміщення) у виконавчу адресу (тобто виконавча адреса визначена у будь-якому режимі адресування).

Оскільки кожне завдання 32-розрядного процесора має максимально 16К ( $2^{14}-1$ ) селекторів, а зміщення можуть бути 4 Гб ( $2^{32}$  біт), то це дає повністю  $2^{46}$  біт або 64 Тб логічного адресного простору на задачу. Цей віртуальний адресний простір доступний для програміста. Блок сегментації перетворює логічний адресний простір у 32-розрядний лінійний. Якщо блока розбиття на сторінки немає, то 32-бітова лінійна адреса відповідає фізичній. Блок розбиття на сторінки перетворює лінійний адресний простір у фізичний. *Фізична адреса* – це те, що є на адресній шині. Головною відмінністю між реальним і захищеним режимом є те, як блок сегментації

виконує перетворення логічної адреси в лінійну. У реальному режимі блок сегментації зсуває селектор ліворуч на 4 біти і додає сформовану в будь-якому режимі адресування *ефективну адресу* до зсуву для утворення лінійної адреси. В захищеному режимі кожний селектор має співвіднесену з ним лінійну базову адресу і зберігається в одній або двох таблицях операційної системи (тобто локальній таблиці дескриптора або глобальній таблиці дескриптора). Лінійну базову адресу селектора додають до зміщення для утворення кінцевої лінійної адреси.

На рис. 2.13 показаний взаємозв'язок між різними адресними просторами.

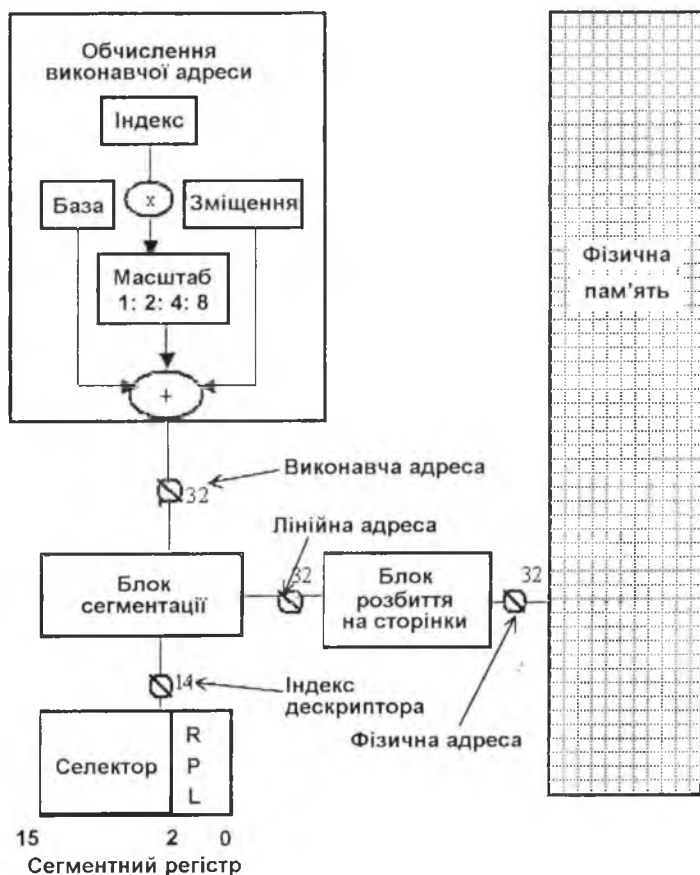


Рис. 2.13. Організація пам'яті. Адресні простори

### Використання сегментного регістра

Як зазначено, головною структурою даних, які використовують для організації пам'яті, є сегмент. *Сегменти* – це різні за розміром блоки лінійних адрес, які мають певні співвіднесені з ними ознаки.

Три головні типи сегментів: коду, даних і стеку, можуть мати розміри від 1 байта до 4 Гб.

Для забезпечення компактного кодування команд і поліпшення робочих характеристик процесора встановлено певні правила використання регістрів. Сегментні регістри вибираються автоматично за замовчуванням відповідно до правил (табл. 2.4).

**Таблиця 2.4. Правила вибору сегментного регістра**

Тип посилання на комірку пам'яті	Використовуваний сегментний регістр (за замовчуванням)	Можливі префікси заміни сегмента (альтернативний)
Вибір коду	CS	Немає
Приймач команд PUSH, PUSHА	SS	Немає
Джерело команд POP, POPА	SS	Немає
Будь-які посилання на дані з адресуванням, що використовує базові регістри:		
[EAX], [EBX], [ECX], [EDX], [ESI], [EDI]	DS	CS, SS, ES, FS, GS
[EBP], [ESP]	SS	CS, DS, ES, FS, GS

Звичайно, посилання на дані використовують селектор, який міститься в регістрі DS. Посилання на дані в стеку застосовують регістр SS і регістр SP як зміщення, а вибирання команд – регістр CS. Вміст вказівника команд IP забезпечує формування зміщення. Спеціальні префікси відміни сегментного регістра дають змогу явно використати заданий сегментний регістр і відмінити неявні правила, перелічені в табл. 2. 4. Префікси відміни дають змогу також застосувати сегментні регістри FS и GS.

Нема жодних обмежень щодо перекриття будь-яких сегментів, тобто сегменти в пам'яті перекриваються довільно. Отже, всі шість сегментів можуть мати базову адресу, що дорівнює нулю, і створити систему з адресним простором у 4 Гб. Це формує систему, в якій віртуальний адресний простір є таким самим, як і лінійний. Детальніше сегментація описана в [7].

### Простір введення/виведення

32-розрядні процесори мають два різних фізичні адресні простори: пам'яті й введення/виведення. Звичайно, зовнішні пристрої використовують простір введення/виведення, хоча 80386 чи 80486 також підтримують інший розподіл пам'яті. Простір введення/виведення складається із 64 Кб і може бути поділений на 64 К 8-бітових портів, 32 К 16-бітових портів, 16 К 32-бітових портів або іншу комбінацію портів. 64 К-адресний простір введення/виведення стосується фізичної пам'яті, оскільки команди введення/виведення не проходять через пристрої, які виконують сегментацію або розбиття на сторінки. Пам'ять введення/виведення працює як додаткова адресна лінія, даючи змогу в такий спосіб легко визначити, до якого адресного простору в конкретний час має доступ процесор. Доступ до портів введення/виведення можна отримати через команди IN (операнд) або OUT (операнд), де операнд містить адресу порту у вигляді безпосереднього операнда (якщо не перевищує 256) або розташований у регістрах DL, DX чи EDX. Усі 8 і 16 біт адреси порту мають нульове розширення (тобто не розширюються) на верхніх адресних лініях. Команди введення/виведення викликають цикл очікування, тому що працюють значно повільніше від мікропроцесора.

Адреси портів введення/виведення від 00F8h до 00FFh зарезервовані для використання числовими співпроцесорами.

### Сегментування

Сегментування – один із способів керування пам'яттю, що забезпечує основу для захисту пам'яті. Сегменти використовують для об'єднання ділянок пам'яті, які мають загальні ознаки. Наприклад, у сегменті можуть міститися всі коди заданої програми або постійно зберігатися таблиця операційної системи. Уся інформація про сегменти зберігається у 8-байтовій структурі даних, яку називають *дескриптором*. Усі дескриптори в системі містяться в таблицях, які розпізнає обладнання. Для керування процесом використання привілейованих інструкцій доступу до дескрипторів застосовують чотирирівневу ієрархічну систему привілеїв. Рівні привілеїв нумерують від 0 до 3. Правила використання рівнів привілеїв контролюють вентилі (Gate) або шлюзи.

Для детальнішого ознайомлення із взаємодією дескрипторів, рівнів привілеїв і захисту варто знати такі терміни:

PL – рівень привілеїв, один із чотирьох ієрархічних рівнів.

Рівень 0 є найпривілейованішим, а рівень 3 – найменш привілейованим. Більш привілейовані рівні за числовим значенням менші, ніж менш привілейовані;

RPL – рівень привілеїв ініціатора запиту (за запитом) – рівень привілеїв оригінального постачальника селектора. RPL визначений найменшими за значенням двома бітами селектора;

DPL – рівень привілеїв дескриптора – найменш привілейований рівень, на якому завдання може вибрати заданий дескриптор і сегмент, співвіднесений із цим дескриптором. DPL визначений бітами 5 і 6 у байті доступу дескриптора;

CPL – поточний рівень привілеїв – рівень, на якому в поточний момент виконується програма; він є рівнем привілеїв сегмента коду, який виконується. CPL може бути також визначений шляхом вивчення двох наймолодших бітів CS регістра, за винятком узгодження сегментів коду;

EPL – ефективний (виконавчий) рівень привілеїв – менш привілейований, ніж RPL і DPL. Оскільки менші значення привілейованих рівнів підкреслюють більший привілей, EPL за числовим значенням є максимальним щодо RPL і DPL.

### **Дескрипторні таблиці**

Таблиці дескриптора визначають усі сегменти, які використовують процесори, починаючи з 80286. У 32-розрядних процесорах є три типи таблиць, які містять дескриптори: таблиця глобального дескриптора (GDT), таблиця локального дескриптора (LDT) і таблиця дескриптора переривання (IDT). Усі таблиці є матрицями пам'яті різної довжини, вони можуть займати від 8 байт до 64 Кб. Кожна таблиця може містити 8192 восьмибайтових дескрипторів. Верхні 13 біт селектора використовують як індекс у таблиці дескриптора. Таблиці мають регістри, співвіднесені з ними, які містять 32-бітову лінійну базову адресу (у 80286 24-бітову фізичну адресу) і 16-бітову межу кожної таблиці. Кожній таблиці відповідає регістр GDTR, LDTR і IDTR (рис. 2.14).

Для локалізації таблиці LDT використовують 16-бітовий регістр LDTR, який містить тільки селектор. Таблиці LDT не є обов'язковими.

Команди LGDT та LIDT завантажують базу і межу GDT і IDT у відповідний регістр. Команди SGDT і SIDT зберігають значення бази і межі в пам'яті. Команда LLDT завантажує сегмент селектора у регістр LDTR, а команда SLDT зберігає селектор у регістрі. Усі таблиці опрацьовує операційна система, отже, команди завантаження таблиці

дескриптора є привілейованими.



Рис. 2.14. Таблиці дескрипторів і відповідні регістри

**Таблиця глобального дескриптора (GDT)** містить дескриптори, доступні для всіх задач у системі. У GDT може бути будь-який тип дескриптора сегмента, крім дескрипторів, які використовують для обслуговування переривань (тобто дескриптори переривання і пастки). GDT є в кожній системі. Звичайно GDT містить код і сегменти даних, які використовують операційні системи, сегменти стану задачі і дескриптори для LDT в системі. Перше гніздо GDT відповідає нульовому селектору, його не застосовують.

**Таблиця локального дескриптора (LDT)** містить дескриптори, які асоціюються (співвідносяться) з заданою задачею. Звичайно, операційні системи створені так, щоб кожне завдання мало окрему LDT. LDT може містити тільки код, дані, стек, вентиль (логічний елемент) задачі і дескриптори вентиля виклику. Таблиці LDT забезпечують механізм для ізоляції коду і сегмента даних заданої задачі від решти операційної системи, тоді як GDT містить дескриптори, загальні для всіх задач. Завдання не може отримати доступу до сегмента, якщо його дескриптора сегмента нема в одній з поточних LDT або GDT. Це забезпечує ізоляцію і захист для сегмента задачі, даючи змогу водночас розділяти глобальні дані між завданнями. На відміну від 6-байтових регістрів GDT або IDT, які містять базові адреси і межі, видима частина регістра LDT містить тільки 16-бітовий селектор, по якому з GDT автоматично завантажуються програмно недоступні поля базової адреси та сегмента.



Таблиці LDT створюються за необхідності.

**Таблиця дескриптора переривань (IDT)** містить дескриптори, які відображають адреси 256 векторів переривань. В IDT можуть бути тільки вентилі задач, переривань і пасток. IDT повинна бути не меншою від 256 байт для того, щоб фіксувати дескриптори для 32 визначених переривань. Кожне переривання, яке використовує система, повинно мати елемент (вхід) в IDT. Елементи IDT аналізують команди INT, вектори зовнішніх переривань і винятки.

### Опис дескрипторів

*Дескриптори* – це структури даних, які використовують для означення властивостей програмних елементів (сегментів, вентилів і таблиць). Дескриптор визначає положення елемента у пам'яті, розмір області, яку він займає (тобто межу), його призначення і характеристики захисту. Дескриптори 16- і 32-розрядних процесорів відрізняються розрядністю поля базової адреси (24 і 32 біти), а також трактуванням поля межі. Детальніше розглянемо структуру дескриптора.

**Атрибутивні біти дескриптора.** Дескриптори є 8-байтовими величинами, які містять атрибути (ознаки) про задану область простору лінійної адреси (тобто сегмента), на який вказує селектор. Ці атрибути включають 32-бітову базову адресу, лінійну адресу сегмента, 20 бітів довжини і ступеня деталізації сегмента, рівень захисту і привілеїв прочитання, написання або виконання, розмір замовчування операндів (16 бітів або 32 біти) і тип сегмента. Вся атрибутивна інформація про сегмент міститься в 12 бітах у дескрипторі сегмента. Звичайний формат дескриптора показано на рис. 2.15.

Усі сегменти в 32-розрядному процесорі мають три загальні атрибутивні поля: P біт, DPL біт і S біт. Біт P (присутність) дорівнює 1, якщо сегмент завантажувється у фізичну пам'ять. Якщо P=0, то будь-які спроби отримати доступ до сегмента викликають виняток неприємності (виняток 2). Рівень привілеїв дескриптора DPL – це 2-бітове поле, яке визначає рівень захисту від 0 до 3, співвіднесений з сегментом. Процесор має дві головні категорії сегментів: системні і несистемні сегменти (для коду і даних). Біт S дескриптора сегмента визначає, чи є заданий сегмент системним, сегментом коду, чи сегментом даних. Якщо S=1, то сегмент є або сегментом коду, або сегментом даних. Якщо S=0, то сегмент є системним.

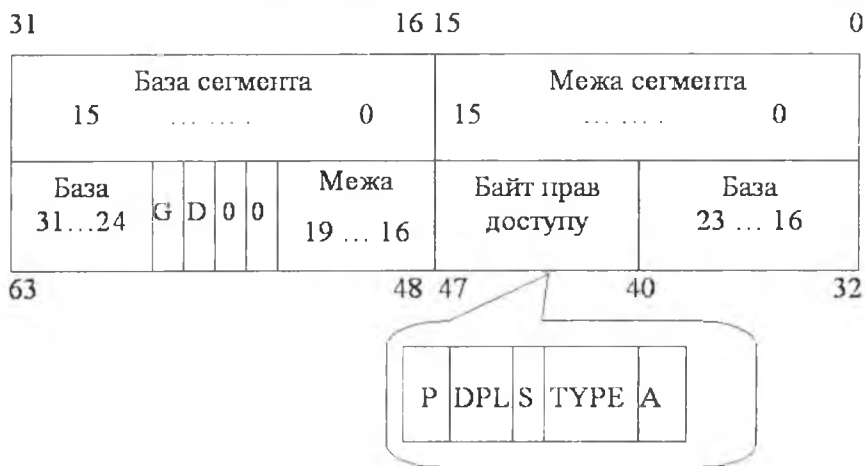


Рис. 2.15. Дескриптори сегмента та розшифрування байта прав доступу

DPL – рівень привілеїв дескриптора; S – сегмент дескриптора; A – біт доступу; P – біт присутності; G – біт гранулярності (G=1 – сторінкова, G=0 – байтова гранулярність); D – біт визначення розміру операнда (D=1–32 біти, D=0–16 біт).

Для реалізації зміни рівнів привілеїв використовуються особливі системні об'єкти – шлюзи, чи вентиля виклику. Вони також мають свої дескриптори.

**Дескриптори сегментів коду і даних (S=1).** На рис. 2.15 зображено загальний формат дескриптора сегментів коду і даних. У цих дескрипторах є декілька загальних полів. Біт A (Accessed – звертання) задається кожного разу, коли процесор отримує доступ до дескриптора. Біт A використовують операційні системи для підтримки статистики застосування сегмента. Біт G (Granularity), або біт ступеня деталізації, визначає, чи є довжина сегмента деталізованою по байтах або по сторінці.

Сегменти процесора можуть мати довжину 1Мб зі ступенем деталізації байта, коли G=0, або 4 Гб, зі ступенем деталізації сторінки, коли G=1, тобто  $2^{20}$  сторінок, кожна довжиною 4 К. Ступінь деталізації зовсім не стосується розбиття пам'яті на сторінки. Система процесора може складатися з сегментів зі ступенем деталізації байтів або зі ступенем деталізації сторінок, незалежно від наявності або відсутності сторінкової організації пам'яті.

Значення бітів поля TYPE для сегмента коду чи даних у порядку зростання позначаються так: R/W, C/ED, E.

Біт виконання E (Executable) визначає, яким є сегмент – коду, чи даних. Сегмент коду, коли  $E = 1$ ,  $S = 1$ , може бути тільки виконаний або тільки прочитаний (запис неможливий), що задано бітом R (Readable – прочитання). Сегменти коду виконуються тоді, коли  $R = 0$ , і зчитуються при  $R = 1$ .

*Зауваження:* Сегменти коду можуть бути модифіковані за допомогою псевдонімів. Псевдоніми – це сегменти даних, які можна записати і які займають ту саму область лінійного адресного простору, що і сегмент коду.

Біт D (Default Operation Size) визначає довжину замовчування для операндів і виконавчих адрес. Якщо  $D = 1$ , то використовують 32-бітові операнди і 32-бітові режими адресування. Якщо  $D = 0$ , то застосовуються 16-бітові операнди і 16-бітові режими адресування. Отже, всі наявні сегменти коду i286 будуть працювати в i386 чи i486, якщо біт  $D = 0$ .

Ще одна ознака сегментів коду визначена бітом C (Conforming – підпорядкованість). Якщо  $C = 1$ , то код може виконуватися, якщо поточний рівень привілеїв (CPL) не нижчий від рівня привілеїв дескриптора (DPL); якщо  $C = 0$ , то керування до заданого сегмента може надатися тільки тоді, якщо  $CPL = DPL$ .

Якщо  $E = 0$ ,  $S = 1$ , то сегменти ідентифіковані як *сегменти даних*, і їх використовують для двох типів сегментів процесора: сегмента стеку і сегмента даних. У цьому випадку біт ED (другий біт поля Type) напрямку розширення визначає, в якому напрямку розглядають сегмент: униз, тобто сегмент стеку, або вгору – сегмент даних. Якщо сегмент є сегментом стеку, то всі зміщення повинні бути більшими, ніж межа сегмента.

У сегментів даних усі зміщення повинні бути меншими або дорівнювати межі. Іншими словами, сегменти стеку починаються з базової лінійної адреси плюс максимальна межа і зростають вниз до базової лінійної адреси плюс межа. З іншого боку, сегменти даних починаються з базової лінійної адреси і розширюються до базової лінійної адреси плюс межа.

Біт W (перший біт поля Type – ознака запису) контролює можливість запису в сегмент. Якщо  $W = 0$ , то сегменти даних можна тільки читати. Сегмент стеку повинен мати  $W = 1$ .

Біт D в останньому байті дескриптора визначає розмір регістра вказівника стеку. Якщо D = 1, то команди PUSH, POP, CALL використовують 32-бітовий ISP-регістр вказівника стеку, і верхня межа набуває значення FFFFFFFFh. Якщо D = 0, то команди, які застосовують стек, використовують 16-бітовий SP-регістр, і верхня межа набуває значення FFFFh.

**Формати дескриптора системи.** Сегменти системи призначені для зберігання локальних таблиць дескрипторів та дескрипторів стану задач і вентилів. Дескриптори 32-розрядних систем містять 23-бітову базову лінійну адресу і 20-бітову межу сегмента.

Дескриптори 16-розрядних систем мають 24-бітову базову лінійну адресу і 16-бітову межу сегмента. Дескриптори i80286 визначені верхніми 16 бітами, які всі дорівнюють нулю.

- Дескриптори LDT (S = 0, TYPE = 2) надають інформацію про локальну таблицю дескрипторів. LDT містять таблицю дескрипторів сегмента, які стосуються тільки певної задачі, бо команда завантаження LDTR можлива тільки на рівні привілеїв 0. Поле DPL ігнорується.

Оскільки таблиці LDT не є обов'язковими, і їх створюють за потребою, то дескриптори LDT містяться в глобальній таблиці дескрипторів GDT. Селектор з регістра LDTR вибирає у таблиці GDT потрібний дескриптор, і атрибути таблиці LDT стають доступними для процесора.

- Дескриптори TSS (S = 0, TYPE = 1, 3, 8, B) сегмента стану задачі містять інформацію про розташування, розмір і рівні привілеїв сегмента стану задачі (TSS). TSS, відповідно, є спеціальним фіксованим сегментом – інформатором, який містить усю інформацію стану задачі і поле зв'язку для можливості вкладення задач.

Поле TYPE використовують для визначення того, чи є завдання поточним, тобто чи воно знаходиться в ланцюжку активних задач, чи в TSS. Поле TYPE також визначає, чи містить сегмент 286 або 386 TSS. У регістрі завдання TR є селектор, який визначає поточний сегмент стану задачі.

- Дескриптори вентилів (шлюзів) (S = 0, TYPE = 4-7, C-F). Вентиль (шлюз) використовують для керування доступом до точок входу в межах сегмента коду. Поле TYPE визначає тип вентиля:

- вентиль виклику – (4)–80286, (C)–80386;
- вентиль задачі – (5)–80286, (D)–80386;
- вентиль переривання – (6)–80286, (E)–80386;
- вентиль пастки – (7)–80286, (F)–80386.

31	16										0									
Селектор										Зміщення 15 ... 0										
Зміщення 31 ... 16										P	DPL	0	TYPE	0	0	0	WORD			

Рис. 2.16. Дескриптори вентилів (32-бітовий формат)

Вентиль забезпечує рівень взаємовідносин між джерелом і адресатом передавання керувань. Це дає змогу процесору автоматично перевіряти захист, а системотехнікам – керувати точками входу в операційних системах. Вентиль виклику застосовують для зміни рівня привілеїв, вентиль задачі – для виконання включення задачі, а вентиля переривання і пастки – для визначення процедури переривання.

Дескриптори вентилів виклику (рис. 2.16) складаються з трьох полів: *байта доступу*, *довгого вказівника селектора* (позначає початок процедури) і ще *слова* (визначає кількість параметрів, яку необхідно скопіювати зі стеку програми, що викликає, в стек процедури, що викликає).

Поле WORD COUNT (поле рахунку слова) використовують у ключах викликів, коли відбувається зміна в привілейованому рівні. Воно визначає кількість слів зі стеку процесу, що викликає, які автоматично копіюються у стек процедури, яку викликають. Інші типи вентилів ігнорують поле WORD COUNT. Вентилі переривання і пастки застосовують селектор і поле зміщення адресата дескриптора вентиля як вказівник початку процедури опрацювання переривань або пастки. Відмінність між вентилями переривання і пастки полягає в тому, що вентиль переривання унеможливорює переривання: скидає біт IF, тоді як вентиль пастки цього не робить. Вентиль задачі використовують для включення задачі, він може стосуватися тільки сегмента стану задачі.

Формат байта доступу є однаковим для всіх дескрипторів вентиля:  $P=1$  означає, що вміст вентиля є істинним,  $P=0$  – що вміст не є істинним і викликає виняток 11, якщо на нього посилаються. DPL – це рівень привілеїв дескриптора, який визначає, коли заданий дескриптор може бути використаний задачею.

**Поля селектора** (рис. 2.17). Селектор у захищеному режимі має три поля:

- 1) індикатор локальної або глобальної таблиці дескриптора TI;
- 2) індекс входу дескриптора;
- 3) рівень привілею ініціатора запиту (селектора).

15	2	1	0
Index (індекс)	TI	RPL	

Рис. 2.17. Формат селектора сегмента

Біти TI позначають, з якої таблиці вибрано дескриптор (TI=0 – глобальна таблиця дескрипторів, TI = 1 – локальна таблиця дескрипторів). Індекс вибирає один з 8K дескрипторів у відповідній таблиці дескрипторів. Біти RPL рівня привілеїв ініціатора запиту не беруть участі у виборі дескриптора, однак їх застосовують для контролю привілеїв у механізмі захисту.

**Кешування дескрипторів сегмента.** Крім значення селектора, кожний регістр сегмента має регістр кеш (cache – буфер, склад) дескриптора сегмента, який з ним співвідноситься. Кожного разу, коли значення регістра сегмента змінюється, 8-байтовий дескриптор, співвіднесений з цим селектором, автоматично завантажується. Вміст кеш-дескриптора є невидимим для програміста. Оскільки кеш-дескриптори змінюються тільки тоді, коли змінюється регістр сегмента, то програми, які змінюють таблиці дескрипторів, повинні перезавантажувати відповідні регістри сегмента після зміни значення дескриптора.

Вміст у кеш-дескрипторах сегмента змінюється залежно від режиму, в якому працює процесор. Для сумісності з архітектурою 8086 базу задають у 16 разів більшою від поточного значення селектора. Межа фіксується на 0000FFFFh, а ознаки – так щоб показати, що сегмент присутній і повністю використовується.

У режимі реального адресування внутрішній рівень завжди встановлюється на найвищий рівень привілеїв, тому команди уведення/виведення та інші операційні коди є привілейованими і виконуються завжди. У захищеному режимі кожне з полів визначене відповідно до вмісту дескриптора сегмента, індексованого значенням селектора, що завантажене в регістр сегмента. Віртуальна програма виконується на найнижчому рівні привілеїв (рівні 3) для того, щоб дозволити організацію пасток усіх його команд, чутливих до PL і команд, які виконуються тільки для рівня 0.

На завершення зазначимо, що механізм сегментування має свої переваги і недоліки. До переваг належать:

- можливість реалізації віртуальної пам'яті;
- побудова операційних систем на базі механізму захисту через привілеї;

- виявлення та опрацювання програмних помилок унаслідок порушення меж чи неправильних вказівників.

Головним недоліком механізму сегментування є необхідність виконувати додаткові службові функції у разі завантаження селекторів у сегментні реєстри. Для цього потрібно мінімум два цикли шини, тому завантаження сегментного реєстра триває значно довше, ніж реєстра загального призначення. Детальніше про це див. [7, 8].

### **Посторінкова організація пам'яті**

**Сутність посторінкової організації пам'яті.** В англomовній літературі для позначення сторінкового типу організації пам'яті використовують термін “підкачування” (swapping), що означає ще один тип керування пам'яттю з метою організації віртуальної пам'яті багатозадачних операційних систем.

На відміну від сегментації, яка організовує програми і дані у модулі різного розміру, підкачування ділить програмні модулі на численні сторінки одного і того ж розміру. Підкачування прямо не стосується логічної структури програми чи даних. водночас селектори сегмента можна розглядати як логічні імена модулів програми або структури даних.

Сторінка звичайно відповідає частині модуля або структури даних. З огляду на переваги локальності коду і посилання на дані в оперативній пам'яті у кожний певний момент часу необхідно тримати лише невелику кількість сторінок з кожної активної задачі.

### **Організація посторінкової пам'яті. Механізм підкачування.**

Механізм керування сторінками використовує дворівневу таблицю для переведення лінійної адреси (з блоку сегментації) у фізичну (рис. 2.18).

Механізм містить три компоненти:

- 1) вказівник (каталог) сторінки;
- 2) таблиця сторінки;
- 3) самі сторінки (рамки сторінки).

Усі елементи резидентної пам'яті механізму підкачування становлять 4 Кб.

Механізм вмикають заданням біта  $PG = 1$  у реєстрі CR0.

Базовий реєстр дескриптора підкачування CR2 є реєстром лінійної адреси помилки підкачування. Він містить 32-бітову лінійну адресу, яка викликає визначення помилки в останній сторінці.

CR3 – це реєстр фізичної базової адреси вказівника сторінки, який містить фізичну початкову адресу цього вказівника. Молодші 12 біт CR3

завжди дорівнюють 0 для того, щоб забезпечити такий стан, за якого вказівник сторінки завжди вирівняний. Завантажують його командою MOV CR3, REG, яка викликає кеш-уведення таблиці сторінок.

**Вказівник сторінки** довжиною 4 Кбайта дає змогу розташувати 1024 одиниць вказівника сторінок. Кожна одиниця вказівника сторінки містить адресу наступного рівня таблиці сторінки та інформацію про ці таблиці. Вміст одиниці вказівника сторінки показано на рис. 2.19. Індексом для вибору правильної одиниці вказівника сторінки слугують старші 10 біт лінійної адреси (A22-A31).

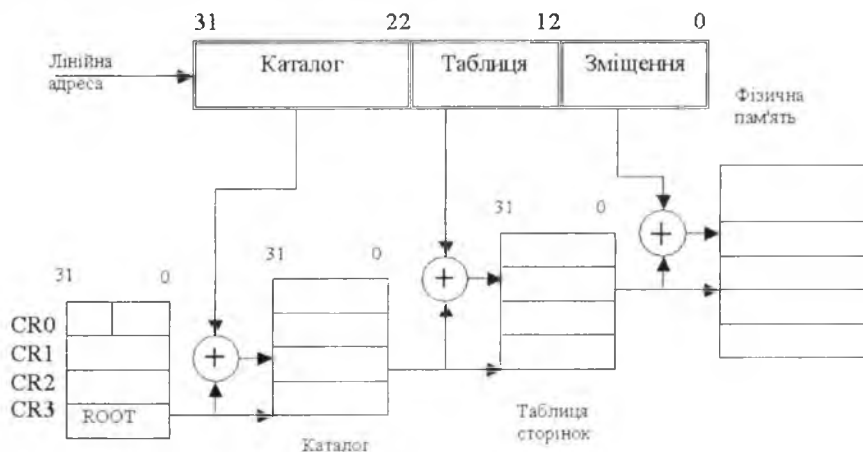


Рис. 2.18. Механізм сторінкової переадресації

**Таблиці сторінок.** Кожна таблиця сторінки має розмір 4 Кб і містить до 1 Кб елементів. Елементи таблиці сторінки містять базову фізичну адресу й атрибути самих сторінок. Індексом пошуку в таблиці одного з елементів використовують біти лінійної адреси A12-A21. Фізична адреса формується з адреси сторінки, яку беруть з таблиці, і молодших 12 бітів лінійної адреси (рис. 2.18).

31	12	11	9	8	7	6	5	4	3	2	1	0
Адреса таблиці сторінок			OS резерв	0	0	D	A	P C D	P W T	U/ S	R/ W	P

Рис. 2.19. Рядок каталога сторінки

Таблиці сторінок можуть бути розташовані між завданнями і записані на диск.



### Елементи таблиці і вказівник сторінок

Молодші 12 бітів елементів таблиці сторінки й елементів вказівника сторінки містять статичну інформацію про сторінки і таблиці сторінки відповідно (див. рис. 2.19). Біт  $P$  визначає, який елемент може бути використаний у передаванні адреси. Якщо  $P=1$ , то елемент можна застосувати для адресної трансляції. Якщо  $P=0$ , то елемент не може бути використаний для трансляції, і всі решта є в наявності для застосування програмою. Наприклад, 37 бітів, які залишилися, можуть бути використані для задання, де на диску розташована сторінка. Біт  $A$  (біт 5, прав доступу) задається процесором для двох типів елементів, перш ніж відбудеться доступ до прочитання або написання перетвореного елемента.

Біт  $D$  (біт 6) дорівнює 1 перед операцією записування за адресою, у перетворенні якої бере участь рядок таблиці, однак він (біт  $D$ ) невинзначений для елементів вказівників сторінки. Біти  $A$ ,  $D$ ,  $P$  зазнають змін процесором апаратно у заблокованих шинних циклах, що запобігає виникненню конфліктів з іншими процесорами або додатковими пристроями. Програми, які змінюють ці біти, повинні використати команду  $LOCK$  для забезпечення цілісності таблиці сторінки для багатористувацьких систем.

Поле  $OS$ -резерв програмно використовується операційною системою під час аналізу того, як довго сторінка була в пам'яті з часу доступу. ОС може виконати алгоритм заміни сторінки як такої, яку останнім часом практично не використовували.

Біт  $PWT$  визначає політику записування у випадку кешування, а біт  $PCD$  забороняє кешування пам'яті для тих сторінок, що обслуговуються (застосовують у процесорах i486).

Біти  $U/S$  (користувач, супервізор) і  $READ/WRITE$  ( $RW$ ) використовують для забезпечення ознак захисту сторінок.

### Режими адресування

Система команд 32-розрядних процесорів забезпечує 11 режимів адресування переходів і даних. Режими адресування оптимізовані, що дає змогу ефективно працювати з такими мовами високого рівня, як  $Ci$ ,  $FORTRAN$  та іншими, крім того, вони охоплюють велику кількість посилань даних, потрібних для мов високого рівня.

### Регістровий і безпосередній режими

Команди, які використовують регістрові і безпосередні операнди, забезпечують два режими адресування:

*Регістровий режим адресування:* операнд розташований в одному з 8-, 16- або 32-розрядному регістрі загального призначення;

*Режим безпосереднього операнда:* операнд уведений в команду як частина операційного коду.

### **Інші режими адресування пам'яті**

Решта дев'ять режимів забезпечують механізм для визначення виконавчої адреси операнда. Лінійна адреса складається з двох компонентів: сегментної базової адреси і виконавчої адреси. Виконавчу адресу обчислюють шляхом додавання будь-якої комбінації таких чотирьох адресних елементів:

**Зміщення:** 8 або 32 біти безпосереднього значення, яке визначене командою (16 бітів зміщення можуть бути використані перед командою з адресним префіксом);

**База:** вміст будь-якого з регістрів загального призначення. Базові регістри звичайно використовує компілятор для задання початку локальної змінної області;

**Індекс:** вміст будь-якого з регістрів загального призначення, крім ESP. Індексні регістри використовують для доступу до елементів масивів або ланцюжка символів;

**Масштаб:** значення індексного регістра може бути помножено на коефіцієнт масштабу 1, 2, 4, або 8. Режим масштабного індексу особливо корисний для доступу до масивів структур.

Комбінації цих чотирьох компонентів становлять дев'ять додаткових режимів адресування. У разі використання будь-якої із цих адресних комбінацій робочі характеристики не погіршуються, оскільки обчислення виконавчої адреси передається по конвеєру з виконанням інших команд. Єдиним винятком є одночасне використання компонентів Базы, Індексу і Зміщення, що потребує додаткового машинного такту (циклу очікування).

Виконавчу адресу (ВА) операнда обчислюють за такою формулою:

$$\text{ВА} = \text{Базовий регістр} + \\ + \text{Індексний регістр} * \text{Масштабний коефіцієнт} + \text{Зміщення}$$

*Прямий режим:* Адреса операнда міститься як частина команди у вигляді 8, 16 або 32 бітів зміщення.

*Приклад:* INC WORD PTR[500], інкремент (вихідний операнд збільшується на 1).

*Регістровий непрямий режим:* Базовий або Індексний реєстр містить адресу операнда.

*Приклад:* MOV [ECX], EDX, пересилання  
(вміст другого операнда пересилається у перший).

*Базовий відносний:* Вміст Базового реєстра додається до зміщення для формування ВА.

*Приклад:* MOV ECX[EAX+4], EDX.

*Індексний режим:* вміст Індексного реєстра додається до зміщення.

*Приклад:* MOV EAX, TABLE[ESI].

*Масштабний індексний режим:* вміст Індексного реєстра множиться на Масштабний коефіцієнт, який додається до зміщення.

*Приклад:* IMUL EBX, TABLE[ESI\*4] цілочислове множення.

*Базовий індексний режим:* вміст Базового реєстра сумується з вмістом Індексного реєстра.

*Приклад:* MOV EAX, [ESI][EBX].

*Базовий індексний масштабний режим:* вміст Індексного реєстра множиться на коефіцієнт масштабу і результат додається до вмісту Базового реєстра.

*Приклад:* MOV ECX, [ESI\*8][EAX].

*Базовий індексний режим зі Зміщенням:* вміст Базового реєстра складається з вмістом Індексного реєстра та зі зміщенням операнда для формування ВА.

*Приклад:* ADD EDX, [ESI][EBP+00FFFFFF0h].

*Базовий індексний масштабний режим зі Зміщенням:* вміст Індексного реєстра множиться на коефіцієнт масштабу і результат додається до вмісту Базового реєстра, який підсумований зі зміщенням.

*Приклад:* MOV EAX, LOCAL TABLE[EDI\*4][EBP+87].

**Відмінності між 16- і 32-розрядними адресами.** З метою забезпечення сумісності з програмами 8086 і 80286, 32-розрядний процесор може виконувати 16-бітові команди в реальному і захищеному режимах.

Мікропроцесор визначає розмір виконуваної команди, аналізуючи біт D у дескрипторі сегмента. Якщо  $D = 0$ , то довжина всіх операндів і виконавчих адрес становить 16 біт. Якщо ж  $D = 1$ , то довжина за замовчуванням для операндів і адрес буде 32 біти. В реальному режимі розмір операндів і адрес за замовчуванням – 16 біт.

Незалежно від визначеної за замовчуванням довжини операндів 32-розрядний процесор здатний виконувати як 16-, так і 32-бітові команди. Цього досягають завдяки префіксу відміни. Два префікси: *префікс розміру операнда* і *префікс довжини адреси*, відмінюють значення біта D на підставі заданого в самій команді значення. Ці префікси автоматично додаються асемблерами INTEL.

*Приклад:* Процесор працює в реальному режимі, і програмісту необхідно отримати доступ до регістра EAX. Команда асемблера для цього може бути такою: MOV EAX, 32bitMEMORYOP (32-розрядний операнд у пам'яті). ASM 386 автоматично визначає необхідність *префікса розміру операнда* і виробляє його.

*Приклад:* Біт  $D = 0$ , і програміст хоче використати масштабний індексний режим адресування, щоб отримати доступ до масиву. Префікс довжини адреси дає змогу використати MOV DX, TABLE[ESI\*2]. Асемблер використовує *префікс довжини адреси* оскільки при  $D = 0$ , за замовчуванням використана 16-бітова адреса.

*Приклад:* Біт  $D = 1$ , і програміст хоче розмістити в пам'ять 16-розрядну величину. Префікс розміру операнда буде використано для визначення тільки 16-бітового значення: MOV 16 біт пам'яті, DX.

### Адресування пам'яті в реальному режимі

Після вмикання живлення або скидання 32-розрядний процесор переходить у *реальний режим* роботи (іноді кажуть, що процесор стає в “нуль”). Цей режим має таку саму базу, як 8086, однак надає доступ до 32-розрядних регістрів. У реальному режимі максимально можна заадресувати 1 Мб пам'яті. Оскільки розбиття пам'яті на сторінки в реальному режимі неможливо виконати, то лінійні адреси є такими ж, як і фізичні. Фізичні адреси формуються в реальному режимі шляхом додавання вмісту відповідного сегментного регістра, який зсувається ліворуч на 4 біти, до виконавчої адреси. Таке додавання приводить до створення адресного простору, який дорівнює 20-бітовій фізичній адресі або 1 мегабайтному адресному простору. Оскільки сегментні регістри зсуваються ліворуч на 4 біти, то це означає що сегменти реального режиму завжди починаються на межах параграфа.

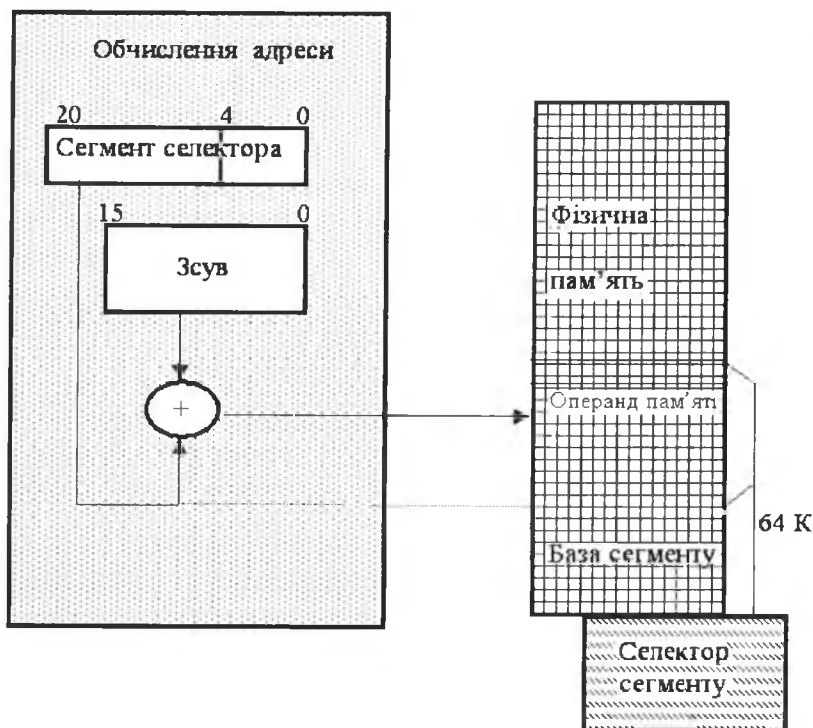


Рис. 2.20. Адресування в реальному режимі

Усі сегменти в реальному режимі мають довжину 64 Кб і можуть бути прочитані, написані або виконані. 32-розрядний МП може генерувати переривання 13, якщо адреса операнда даних або виклик команди відбуваються після кінця сегмента. (Тобто адреса операнда має зміщення більше ніж FFFFh, або, наприклад, слово з молодшим байтом з адресою FFFFh та старшим байтом з адресою 0000h.)

Сегменти в реальному режимі можуть перекриватися, тобто якщо певний сегмент не використовує всі 64 Кб, то інший сегмент може бути накладений на невикористану частину попереднього сегмента. Це дає змогу програмісту зменшити кількість фізичної пам'яті, необхідної для програми.

#### Запасні комірки пам'яті

Є дві фіксовані області в пам'яті, які резервовані в режимі реального адресування:

- 1) область ініціалізації системи;
- 2) область таблиці переривання.

Комірки від 00000h до 003FFh резервовані для векторів переривань. Кожне з 256 можливих переривань має зарезервовану 4-байтову адресу переходу.

Комірки від FFFFFFF0h до FFFFFFFFh резервовані для ініціалізації системи.

### 2.1.1.5. Захист та привілеї

Функція захисту процесора полягає у забороні несанкціонованого виконання критичних команд, зокрема, команди HLT, яка зупиняє процесор, а також команд, що впливають на сегменти коду і даних. Розрізняють три групи механізмів захисту:

- обмеження використання сегментів;
- обмеження доступу до сегментів через застосування правил привілеїв;

- наявність привілейованих команд або операцій, які можуть бути виконані тільки за певних рівнів CPL та IOPL.

На відміну від традиційних систем, які використовують у мікропроцесорах для захисту, 80386 забезпечує захист як частину інтегрального блоку керування пам'яттю і не використовує складного зовнішнього обладнання та програмного забезпечення. 32-розрядні процесори пропонують також додатковий вид захисту, що ґрунтується на сторінковій організації пам'яті.

Тридцятидворозрядні процесори у захищеному режимі мають чотири рівні привілеїв, які оптимізуються для підтримки потреб багатозадачних ОС, ізолюють і захищають програми користувача одну від одної і від ОС.

Рівні привілеїв керують використанням привілейованих команд, а також командами введення/виведення і доступом до сегмента і дескриптора сегмента. У разі виконання майже кожної машинної команди відбувається перевірка захисту.

Рівні привілеїв пронумеровані від 0 до 3. Рівень 0 є найбільш привілейованим (рис. 2.21).

Традиційні системи супервізор/користувач працюють з двома кільцями захисту. Переважно супервізору (тобто операційній системі) відведено кільце 0, а користувачеві – кільце 3 (наприклад, ОС UNIX).

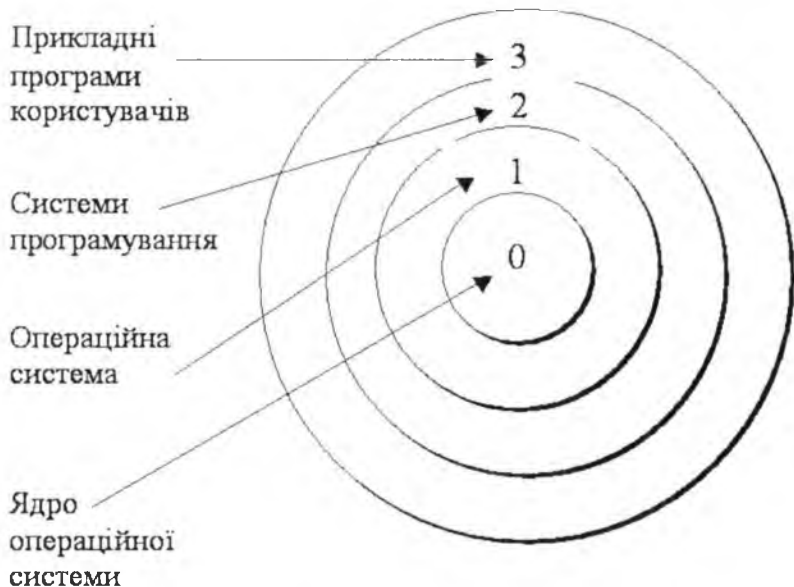


Рис. 2.21. Рівні привілеїв, або кільця захисту

### Правила привілеїв

Процесор керує доступом як до даних, так і до процедур між рівнями привілеїв задач згідно з такими правилами: дані, які поміщені в сегмент з рівнем привілеїв  $P$ , можуть бути доступні тільки з кодом, що працює з рівнем привілеїв у крайньому разі з таким же ступенем привілеїв, як  $P$ .

Сегмент, або процедура коду, з рівнем привілеїв  $P$ , може бути викликана задачею, що працює на тому ж або менш привілейованому рівні, ніж  $P$ .

### Рівні привілеїв

**Привілеї задач.** У будь-який момент часу 32-розрядний процесор виконує завдання у одному з чотирьох рівнів привілеїв. Поточний рівень привілеїв CPL (встановлюється двома молодшими бітами регістра CS) визначає рівень привілеїв задачі. CPL задачі можна змінити тільки процедурою керування через дескриптори вентиля до кодового сегмента з іншим рівнем привілеїв. Отже, програма, яка виконується на рівні привілеїв 3, може викликати програму ОС на рівні привілеїв 1 засобами вентиля. Задача, яка виконується на нульовому рівні привілеїв, має доступ до всіх сегментів, які описані у GDT, і є найбільш привілейованою.

**Привілеї селектора RPL.** Рівень привілеїв селектора визначений двобітовим полем RPL селектора, і його використовують тільки для задання меншого рівня привілеїв у порівнянні з поточним рівнем привілеїв для заданого сегменту.

Цей рівень називають ефективним рівнем привілеїв задачі EPL і визначають як найменш привілейований рівень CPL задачі і RPL селектора, тобто якщо RPL селектора дорівнює 0, то CPL завжди визначає рівень привілеїв для можливого доступу використання селектора. Якщо ж  $RPL = 3$ , то доступ до сегмента на рівні 3 відбудеться незалежно від CPL задачі.

Селектор, що його формує програма, може обмежити будь-яке значення RPL. Це забезпечує команда налагодження RPL – ARPL, яка змушує втручатися в біти RPL, CPL постановника.

**Привілеї введення/виведення.** Рівень привілеїв введення/виведення IOPL дає змогу коду операційної системи, що працює при  $CPL = 0$ , обмежити менш привілейований рівень, за якого можуть бути використані команди введення/виведення.

Виняток 13 – порушення загального захисту – виникає тоді, коли застосовують команду введення/виведення, а CPL задача менш привілейована, ніж IOPL (пригадаємо, що IOPL розташовується в бітах 13, 14 регістра EFLAGS).

Команди INT, INC, OUT викликають виняток 13, якщо CPL більше, ніж IOPL.

Для того, щоб задачі “оминали” спрацьовування захисту, процесор має декілька спеціальних команд перевірки вказівника, що допомагає підтримувати цілісність системи.

Команда ARPL {операнди, селектор, регістр, функція} налагоджує потрібний рівень привілеїв: APL селектора, числовий максимум поточного селектора, значення RPL і значення RPL у регістрі; встановлює нульову ознаку  $ZF = 1$ , якщо селектор RPL був змінений.

Команда VERR {операнди, селектор, функція} – перевірка для читання (верифікація читання) – задає нульову ознаку, якщо сегмент, до якого належить селектор, може бути прочитаний.

Команда VERW {операнд, селектор} – перевірка для запису (верифікація запису) – задає нульову ознаку, якщо сегмент, на який посиляється селектор, може бути записаний.



Команда LSL {операнди, регістри, селектор, функція} завантажує межу сегмента, зчитуючи її в регістри, якщо правило привілеїв і тип дескриптора це дозволяють.

Команда LAR {операнд, регістр, селектор, функція} виконує завантаження прав доступу. Зчитує байт права доступу дескриптора в регістр, якщо дозволяють правила привілеїв. Ознака IF=1 в разі успішного завершення обох останніх команд.

Процесор виконує перевірку захисту. Спочатку він перевіряє, чи сегмент не є нульовим. Неправильний сегмент виробляє виняток 11, після того перевіряє, чи належить селектор до правильного типу сегмента.

Контроль доступу до сегментів даних здійснюється у разі виконання команд завантаження селекторів у регістри SS, DS, ES, FS, CS. Ці команди, окрім завантаження у SS, змушені робити посилання на дескриптори сегментів даних або сегментів команд. Винятком з цих правил є дозволений для читання підпорядкований кодовий сегмент, який може бути прочитаний з будь-яким значенням CPL. Після того врешті-решт виконується перевірка правильності привілеїв. CPL порівнюється з EPL і, якщо EPL більше привілейований, ніж в CPL, то виникає виняток 13.

Правила, які стосуються сегмента стеку, дещо відрізняються від тих, які втягують сегменти даних. Команда, яка завантажує селектор SS, повинна посилатися на дескриптор сегмента даних, що дозволяє записування. Для цього DPL і RPL повинні дорівнювати CPL. Порушення цієї умови або порушення правил привілеїв вироблятимуть виняток 13.

Помилка – стек не присутній – також виробляє виняток 13, а для для несутнього коду або сегмента даних характерний виняток 11.

**Передавання рівня привілеїв.** Передавання керування всередині сегмента відбувається тоді, коли селектор завантажується в регістр CS. Для звичайних систем більшість з цих передавань є просто результатом виклику або переходу до інших програм.

Є п'ять типів передавання керувань. Частина з них приводить до передавання рівня привілеїв. Зміна рівня привілеїв виконується тільки внаслідок передавання керування з використанням вентилів включення задачі і переривання або вентилів пасток. Передавання керувань може відбутися тоді, коли селектор співвідноситься з правильним типом дескриптора. Будь-яке порушення використання правил дескрипторів спричинить виняток 13.

Для безпеки системи всі передавання керувань підпорядковані таким правилам привілеїв:

1) переходи рівня привілеїв відбуваються тільки через вентиля (шлюзи);

2) команди JMP і CALL можуть здійснювати посилання або на підпорядкований сегмент коду з DPL, який є більшим або дорівнює CPL, або на непідпорядкований сегмент з DPL, який дорівнює CPL.

3) переривання, опрацювання яких відбувається в межах задачі, підпорядковані також правилам привілеїв, як і CALL;

4) сегменти підтвердження коду доступні для рівнів привілеїв, які є такими ж або менш привілейованими, ніж CS сегмента підтверджувального коду. Потрібний рівень привілею RPL в селекторі, що визначає вентиль і CPL задачі, повинні дорівнювати або мати більший привілей, ніж CPL вентиля;

б) переривання, які можуть змінити рівень привілеїв, можуть передавати керування сегменту коду з рівнем привілеїв, таким же або більш привілейованим, ніж CPL, тільки через вибраний вентиль з таким же або більш привілейованим, ніж CPL задачі;

7) команди повернення, які не перемикають задачі, можуть повернути керування тільки до сегмента коду з таким же або меншим привілеєм.

Зміна рівня привілеїв, яка відбувається в разі передавання керувань, приводить до автоматичного перевизначення стеку. Увімкнення задачі повинно бути виконано командами CALL, JMP або INT, які належать або до вентиля задачі, або до сегмента стану задачі, чий DPL менший або мають такий самий привілей, як і CPL попередньої задачі. У разі повернення на попередній рівень привілеїв його стек відновиться як частина IRET- або RET-команди.

#### **2.1.1.6 Кешування пам'яті у персональних комп'ютерах**

За означенням кеш-пам'ять є меншої ємності, ніж оперативна, отже не може зберігати копію всієї оперативної пам'яті. Цей тип пам'яті зберігає лише обмежену кількість інформації і таблицю (список) відповідності даних ділянкам основної пам'яті. Крім того, не вся оперативна пам'ять, яка доступна процесору, може кешуватися. Головною причиною цього є можливості контролера кешу.

Ефективність роботи кеш-пам'яті очевидно залежить від того, наскільки вдало розташовані в ній дані, до котрих відбуваються звернення процесора. Розрізняють два випадки:

1) якщо в результаті звернення процесора до кешу знайдено відповідні дані, попередньо прочитані з основної пам'яті, то вважають, що відбулося *кеш-попадання* (cache hit);

2) якщо в результаті звернення процесора даних у кеші не виявилось, то вважають що відбувся *кеш-промах* (cache miss). У цьому випадку процесор повинен прочитати дані з основної пам'яті.

Відношення кількості кеш-попадань до загальної кількості звернень називають коефіцієнтом збігу, або успіху. Відсоток вдалих попадань головно залежить від алгоритму кешування блоків даних з основної пам'яті до кешу.

Контролер кешу забезпечує передавання рядків даних певної довжини (cache line). Кожному рядку кешу відповідає певний блок даних основної пам'яті та інформація про адресу скопійованих у нього даних і про її стан. Якщо в поточний момент часу в рядку відображена достовірна інформація, то такий рядок називають дійсним (valid), у протилежному випадку – недійсним. Інформація про адресу блока даних чи номер сторінки та стан рядка називають тегом (tag) і зберігають у пов'язаній з даним рядком комірці спеціальної пам'яті тегів (tag RAM). Можливий варіант секторованого кешу, в якому один рядок містить інформацію про кілька суміжних комірок (секторів).

Розрізняють дві політики чи стратегії запису даних з кешу в оперативну пам'ять: наскрізний запис WT (Write Through) і зворотний запис WB (Write Back). Наскрізний запис передбачає виконання кожної операції запису одночасно в рядок кеша й оперативну пам'ять. Ця стратегія застосована у перших процесорах i486. У сучасних процесорах переважає стратегія зворотнього запису, суть якої полягає у зменшенні кількості операцій запису на системній шині основної пам'яті. Докладніше про це див. у [16, 25].

Залежно від способу відображення блоку основної пам'яті на рядок кешу розрізняють три типи архітектур кеш-пам'яті:

- кеш прямого відображення (direct-mapped cache);
- повністю асоціативний кеш (fully associative cache);
- набірно-асоціативний кеш (set-associative cache).

Кеш прямого відображення передбачає, що адреса пам'яті, за якою відбувається звернення до кеша, однозначно визначає рядок кеша, де може знаходитися відповідний блок. Кеш цього типу застосовується у вторинному кеші більшості системних плат сучасних ПЕОМ. У цьому типі архітектури легко обчислити ємність кешованої основної пам'яті  $M_{\text{кеш}}$

$$M_{\text{кеш}} = V_{\text{с}} \cdot 2^n,$$

де  $V_{\text{с}}$  – ємність кеш-пам'яті,  $n$  – розрядність пам'яті тегів.

Наприклад, якщо на системній платі встановлено кеш ємністю 256 Кб і розміром рядка 32 байти, то ємність кешованої пам'яті дорівнює 64 Мб.

У повністю асоціативному кеші будь-який його рядок може відображати будь-який блок основної пам'яті. Він реалізований для обмеженого числа кешів першого рівня.

Набірно-асоціативний кеш містить кілька паралельних і узгоджених працюючих каналів прямого відображення. Ця архітектура широко застосовується для первинного кешу сучасних ПЕОМ. Ємність кешованої пам'яті визначається так само, як і у випадку прямого відображення, однак тут розрахунок ведеться для одного блока, а не всього кешу.

Розглянемо роботу чотириканальної набірно-асоціативної кеш-пам'яті на прикладі **внутрішньої кеш-пам'яті i486**. КП несекторована, тобто кожний біт достовірності стосується цілого рядка. Якщо адреса потрапляє у кеш, то з нею порівнюють усі теги і у випадку зрівняння одного з тегів процесор запитує інформацію за адресою, що асоціюється з тегом (потрапляння). У цьому випадку цикл шини не потрібний. Якщо інформації у КП нема (випадок промаху), то виконується заповнення рядка КП за одне або декілька 16-байтових передавань даних (для процесорів Pentium – 32-байтові). Схема організації внутрішньої асоціативної кеш-пам'яті показана на рис. 2.22.

Адресування у КП відбувається шляхом розділення старших 28 бітів фізичної адреси на три частини: 7 бітів поля індекса визначають номер множини (із 128 множин). Старші 21 бітів є полем ознаки, або тегу. Ці біти порівнюються з тегами кожного рядка в індексованій множині і відображають, чи зберігається 16-байтовий рядок КП за заданою фізичною адресою. Молодші 4 біти фізичної адреси вибирають байт усередині рядка КП.

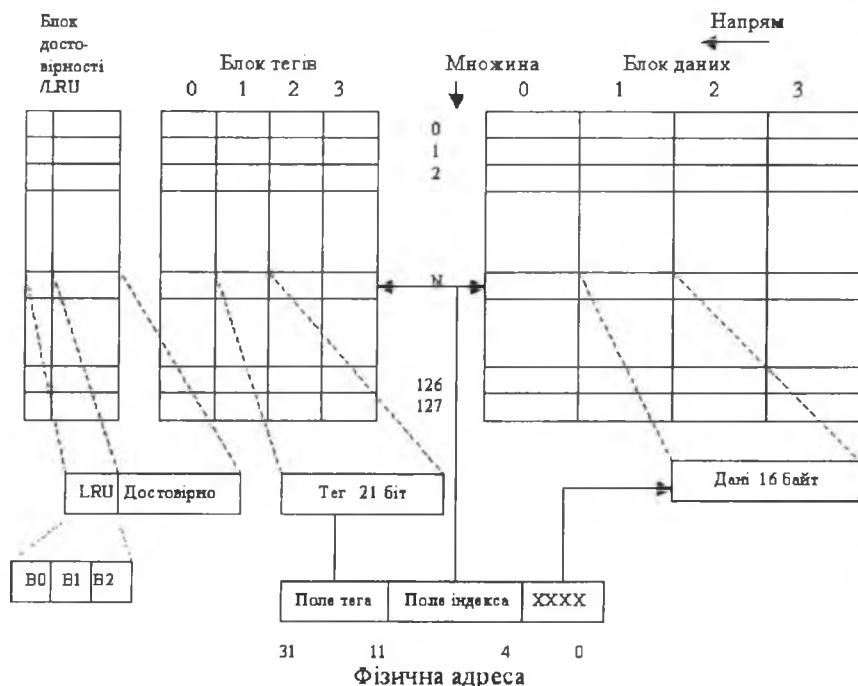


Рис. 2.22. Схема організації внутрішньої кеш-пам'яті

КП функціонально розділена на три блоки: блок даних, блок тегів та блок достовірності і LRU (Least Recently Used – той, що використовувався найдавніше). У цьому випадку застосовано алгоритм псевдо LRU, який описується згодом.

Блок даних містить до 8 Кб і розділений за чотирма напрямками, кожний з яких має 128 16-байтових множин, або рядків КП. Такий вибір є компромісом між швидкодією КП з прямим відображенням у разі потрапляння і великим коефіцієнтом потраплянь повністю асоціативної пам'яті.

Режим роботи КП визначений програмно (CR0, біт CD). Унаслідок очищення КП всі біти достовірності дорівнюють нулю.

**Алгоритм псевдо LRU:** біти LRU оновлюються під час кожного “потрапляння” в КП або заповнення рядка. Біти достовірності в разі очищення КП дорівнюють нулю. Якщо у циклі виявиться “промах” в КП, і потрібно передати з пам'яті певний рядок, то для заповнення буде

вибрано один з чотирьох рядків множини. Якщо у множині є недостовірний рядок, то власне він і заповниться. Якщо ж усі рядки достовірні, то замінюваний рядок буде вибрано за допомогою бітів з LRU.

*Схема вибору заміни рядка.* Якщо всі рядки в множині достовірні, то вибір рядка для заміни відбувається за такою схемою:

B0	B1	B2	
0	0	x	змінюється рядок L <sub>0</sub>
0	1	x	змінюється рядок L <sub>1</sub>
1	x	0	змінюється рядок L <sub>2</sub>
1	x	1	змінюється рядок L <sub>3</sub>

Модифікація бітів LRU відбувається так:

- якщо останнє звертання в множині було до рядка L<sub>0</sub> або L<sub>1</sub>, то біт B0=1, а в разі звернення до рядка L<sub>2</sub> або L<sub>3</sub> біт B0=0;
- якщо останнє звернення в парі L<sub>0</sub>-L<sub>1</sub> було до рядка L<sub>0</sub>, то B1=1, а в разі звернення до L<sub>1</sub> B1=0;
- якщо останнє звернення в парі L<sub>2</sub>-L<sub>3</sub> було до рядка L<sub>2</sub>, то B2=1, а в разі звернення до L<sub>3</sub> B2=0.

У випадку потраплянь КП працює з такою ж швидкістю, як регістри процесора.

У МП 80486 та наступних процесорах за допомогою пакетних циклів (в одному пакеті довжиною до 16 байт можна передавати тільки вирівняні блоки даних за суміжними адресами) подвійне слово з пам'яті зчитується за кожний такт. Для пересилання одного рядка кешу потрібно чотири 32-розрядних шинних цикли. Завдяки цьому дуже швидко заповнюється внутрішня КП і буфер попереднього вибирання команд. У результаті використання КП близько 90-95% запитів на читання задовольняє власне вона і тільки 5-10% припадає на промахи. Про пакетний режим передавання даних можна довідатись детальніше у [7, 8].

### 2.1.1.7. Віртуальна конфігурація 8086

32-розрядний процесор дає змогу виконувати програми 8086 як у реальному, так і у віртуальному режимах. Віртуальний режим 32-розрядного процесора (V86) для користувача є гнучкішим і привабливішим, оскільки можна скористатися всіма перевагами цього процесора і, крім того, виконувати в ньому програми, як і в реальному режимі.

Механізм захисту процесора забезпечує одночасне виконання операційної системи і програм 8086, а операційна система мікропроцесора 80386 – також програм 80286, 80386.

Усі програми, які виконуються у V86, мають рівень привілеїв 3. Адресування відбувається так само, як і в реальному режимі. Увійти у режим V86 (біт регістра ознак VM = 1) можна двома способами:

- виконати команду IRET у 32-розрядному режимі, коли копія регістра ознак збережена у стеку з заданим бітом VM (при CPL = 0);
- перемкнути задачу на 32-розрядну, в якій в TSS копія регістра ознак має заданий біт VM.

Вихід з режиму V86 можливий тільки в разі опрацювання переривання. Новий режим (захищений чи V86) буде задано відповідно до TSS нової задачі.

### **Механізм адресування у віртуальному 8086 режимі**

Однією з головних відмінностей адресування віртуального режиму від решти (реального і захищеного) є те, як інтерпретовано селектори.

У віртуальному 8086 режимі, як і в реальному, для отримання лінійної адреси вміст сегментних регістрів зсувається ліворуч на 4 біти і додається зміщення. Завдяки посторінковій організації пам'яті адресний простір задачі в 1 Мб у віртуальному режимі може бути відображений у будь-якому місці 4-гігабайтового простору лінійної адреси 32-розрядного процесора.

У віртуальному режимі адреси, які перевищують 1 Мб, будуть виробляти виняток 13 (як і в реальному режимі), однак ці обмеження не є важливими, оскільки більшість задач, які виконуються у віртуальному 8086 режимі, є одиничними програмами мікропроцесора 8086 (а їхній розмір не перевищує 1 Мб).

Механізм сторінкової адресації дає змогу одночасно опрацьовувати численні програми і забезпечує захист та ізоляцію операційної системи. Завдяки обладнанню свопінгу можна 20-бітову лінійну адресу, яку виробляє програма у віртуальному режимі, розбити на 256 сторінок. Одна зі сторінок може бути розташована у будь-якому місці в межах 4-гігабайтового простору фізичної адреси 32-розрядного процесора. Крім того, оскільки CR3 (базовий регістр вказівника сторінки) завантажується у разі перемикання задачі, то кожне завдання у віртуальному режимі може використати різні схеми для того, щоб відобразити сторінки на різні місця в пам'яті. І, нарешті, обладнання

підкачування дає змогу розділити код ОС 8086 між численними аплікаціями 8086.

### 2.1.1.8. Тестування. Самоконтроль

У 32-розрядних процесорах розвинені засоби тестування і налагодження. Виконувати самоконтроль можуть процесори, починаючи з 80386. Самоконтроль перевіряє функціонування всіх пристроїв постійної пам'яті і більшість нерегулярної логіки частини машини. Під час самоконтролю може бути перевірена приблизно половина мікропроцесора. У 80386 самоконтроль ініційований тоді, коли відбувається перехід від високого сигналу до низького на виводі Reset, а на виводі BUSY низький сигнал. Самоконтроль займає приблизно 30 мілісекунд для процесора з тактовою частотою 16 МГц. Після закінчення самоконтролю процесор виконує скидання і починає звичайну роботу. Елемент вважається таким, що успішно пройшов контроль, якщо вміст регістрів EAX і EDX дорівнює 0. Якщо ж вміст EAX і EDX відрізняється від нуля, то це означає, що в процесі самоконтролю було виявлено дефект в елементі.

Регістр DH після скидання містить інформацію про тип процесора, а регістр DL – номер моделі.

З деяких моделей 80486 у процесори почали вводити підтримку тестового інтерфейсу JTAG, який відповідає стандарту IEEE 1149.1 Boundary Scan Architecture. Тестуванню підлягає внутрішня логіка процесора за методом сканування меж. Цей інтерфейс є і в процесорах типу Pentium, де він удосконалений і надає доступ до регістрів процесора.

### Тестування буфера асоціативної трансляції

Мікропроцесор 80386 має механізм для контролю буфера асоціативної трансляції (TLB). Ця можливість потрібна передусім тим, хто пише тестові програми для мікропроцесора. Тестування TLB є унікальною властивістю 80386 і в майбутніх процесорах можуть не використовувати. Тестування TLB потребує застосування тестера і програми мовою асемблера для надання дії моделі тесту. Сторінкова організація пам'яті в цьому разі повинна бути заблокована.

Два тестові регістри (див. рис. 2.23) забезпечені засобами написання зразка в TLB і зчитуванням результату. TR6 є регістром тестових команд, а TR7 – контрольних даних.



31	12 11										0
Лінійна адреса	V	D	D	U	U	W	W	0	0	0	0
Фізична адреса	0	0	0	0	0	0	0	PL	REP	0	0

Рис. 2.23. Тестові регістри TR6 та TR7

Тестові регістри дають змогу проводити два види операцій з TLB: *напиши нове TLB введення; виконай шукання TLB.*

Запис у регістр тестової команди через команду MOV TR0, (регістр) приводить до виконання операції TLB. Якщо біт 0 дорівнює 1, то виконуватиметься шукання TLB.

### Забезпечення налагодження

32-розрядні процесори мають розвинені засоби налагодження. Більшість цих засобів призначені передусім для програмного налагодження. (INTEL забезпечує повний набір інструментів налагодження, таких як ICE-386 (внутрішньосхемна емуляція) і PTM-386 (переглядовий монітор) для закінчення побудови налагоджувальних ознак).

Три головні типи засобів налагодження мікропроцесора такі:

- точки зупинки програмного забезпечення;
- покроковий режим роботи;
- регістри налагодження.

Однобайтове переривання INT3 відладники програмного забезпечення використовують для реалізації точок зупинки. Після того, як команду INT3 введено й опізнано, виконання продовжиться на опрацювачі переривання 3. Одноступенєвого переривання досягають шляхом задання біта (TF) в регістрі EFLAGS. TF біт задають шляхом зміни копії регістра ознак у стеку і виконанням команди POPF або IRET. У разі задання біта TF одноступенєве переривання відбудеться після виконання наступної команди. Команда переривання помістить у стек поточний регістр ознак (з заданим бітом TF), а після того очистить цей біт (забезпечуючи можливість нормального виконання програми опрацювання одноступенєвого переривання). Це дає змогу створити опрацювач переривання, який виконуватиме ступеневі рухи всередині команди. Одноступенєве переривання використовує вектор переривання 1, який надходить до процесора внутрішньо. Після завершення програми опрацювання одноступенєвого переривання команда IRET добує регістр ознак зі стеку і передасть керування наступній команді, яка також буде розбита на ступеневі кроки.

На відміну від традиційних точок зупинки, які підтримують тільки команди виклику переривань, регістри налагодження дозволяють задавати точки зупинки для доступу до даних. Отже, якщо змінна величина випадково перезаписана (або виконано накладення), то точка зупинки може бути задана, щоб припинити виконання незалежно від того, чи буде змінено значення цієї змінної. На рис. 2.24 показані регістри налагодження. DR0-3 містять лінійні адреси точок переривання.

Лінійна адреса може не відповідати адресі фізичній, якщо розблокована посторінкова організація пам'яті. DR6 містить стан регістрів точок зупинки. Біти в регістрах мають такі значення:

- біт BT: задано, якщо відбувається введення задачі в задачу, де TSS має заданий біт пастки налагодження;
- біт BS: дає змогу опрацюювачам налагодження розрізняти одноступеневі пастки від інших умов налагодження;
- біт BD: заданий обладнанням, якщо наступна команда отримує доступ до регістра налагодження;
- біти B0-B3: ці біти задані, якщо відбулося переривання обмеженого використання. B0 задано, якщо відбулося переривання 0 (точка зупинки) і т. д.:

Лінійна адреса точки зупинки 0																																DR0																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
Лінійна адреса точки зупинки 1																																DR1																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
Лінійна адреса точки зупинки 2																																DR2																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
Лінійна адреса точки зупинки 3																																DR3																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
Резерв. Значення бітів невизначені																																DR4																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
Резерв. Значення бітів невизначені																																DR5																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Рис. 2.24. Регістри засобів налагодження

DR7 – регістр керування налагодженням. Його використовують для дозволу або уточнення різних точок зупинки. Біти мають такі значення:

LEN – це двобітове поле, яке визначає довжину переривання. Всі переривання повинні бути вирівняні: двобайтові вектори – по межах слова; чотирибайтові – по межах подвійних слів. Значення поля: 00 – довжина в один байт, 01 – довжина в один байт, 10 – невизначена, 11 – довжина в чотири байти.

RWE: це двобітове поле визначає тип вибірки з пам'яті, яка повинна відбутися для того, щоб активізувати опрацювання переривання:

Поле RWE	Тип вибірки
00	Тільки виконання команди
01	Тільки записування даних
10	Невизначений
11	Тільки зчитування та записування даних (команди не вибираються)

GE/LE – глобальна і локальна верхні точки зупинки: ці біти повинні завжди дорівнювати 1 у разі використання переривань.

Gi/Li – розблокування глобальної і локальної точок зупинки: якщо або Gi=1, або Li=1, то переривання розблоковуються. Якщо ці біти задані, то будь-яке переривання обмеженого використання (тобто точка зупинки, яка відповідає умовам, що визначені бітами LWE) змусить процесор виконувати програму опрацювання налагодження. Біти Li дають змогу задавати локальні точки зупинки для індивідуальної задачі, не впливаючи на іншу задачу. Gi біти допомагають задавати переривання, які діють на всі задачі.

Для того, щоб задати точку зупинки, мікропроцесор повинен працювати на нульовому рівні привілеїв у реальному режимі. Після того точку зупинки потрібно задати шляхом завантаження регістра точки переривання (засобом MOV DRi – операнд в пам'яті або регістрі команди) з адресою точки зупинки. Згодом потрібно задати відповідний LEN та RWE, і нарешті, біти розблокування точки переривання Gi і/або Li.

Молодші чотири біти Bi в DR6 стосуються чотирьох апаратних контрольних точок; одиничні стани відобразять досягнення точки зупинки, лінійна адреса якої міститься у регістрі DRi. Однак доти, доки не будуть задані Gi або Li, процесор не виконуватиме програми налагодження з цими перериваннями (детальніше див. [8]).

Генерування винятків за контрольними точками можна заборонити заданням ознаки RF у регістрі ознак.

### 2.1.1.9. Режим системного керування SMM

Сучасні моделі 32-розрядних процесорів (починаючи з деяких модифікацій 386 і 486-го) мають додатковий режим системного керування SMM (System Management Mode). Головно його використовують

для реалізації системи керування енергоспоживанням. Відразу після входження у режим процесор зберігає свій стан (контекст) – майже всі регістри – у спеціальній ділянці фізичної пам'яті SMRAM (розмір від 32 Кб до 4 Гб). Згодом процесор переходить до виконання опрацювача SMI, який міститься також у SMRAM. У цьому випадку заборонено всі типи переривань і винятки не генеруються. Процедура опрацювача завершується командою RSM, внаслідок виконання якої процесор відновлює свій стан з копії, яка зберігалася у SMRAM. Якщо не задано спеціального режиму, то регістри стеку і кодів операцій мають 16-бітове значення. Режим SMM має такі властивості:

- обчислення адрес відбувається так само, як і реальному режимі;
- межа не перевищує значення 4 Гб;
- ознака переривань IF=0;
- немасковане переривання NMI заборонене;
- ознака TF у регістрі EFLAGS не задана (покроковий режим заборонений);
- регістр DR7 занулений, пастки для налагодження заборонені;
- може бути виконана команда RSM (у інших режимах вона викликає неправильний код операції).

Значення базової адреси SMBASE (переважно 3000h) задає процесор у випадку скидання (сигнал RESET), її можна змінити програмним способом унаслідок дії опрацювача SMI. Для цього достатньо змінити значення базової адреси у ділянці збереженого контексту за адресою SMBASE+FEF8h і тоді після виконання команди RSM під час опрацювання сигналу SMI буде використана нова ділянка пам'яті.

Для керування рестартом команд уведення/виведення застосовують слово за адресою SMBASE+FF00h. Рестарт команди буде виконано в тому випадку, коли опрацювач запише у слово код 0FFh, а якщо нульове значення слова не буде змінено, то після виходу з SMM процесор виконуватиме наступну команду. Можливий рестарт команди HALT (зупинка процесора), якщо додано нульовий біт слова за адресою SMBASE+FF02h. Цей біт встановлюється в одиничний стан у випадку появи переривання під час виконання команди HALT. Якщо опрацювач SMI збереже це значення, то після завершення SMM процесор повторно виконає команду HALT, а якщо не збереже, то процесор виконає команду, яка є наступною після HALT (детальніше див., напр. [8]).

### 2.1.1.10. Програмна архітектура співпроцесора

Співпроцесор підтримує сім типів даних: 16-, 32-, 64-бітні цілі числа; 32-, 64-, 80-бітні числа з плаваючою комою і 18-розрядні числа у двійково-десятковому форматі. З програмної точки зору співпроцесор і цілочисловий процесор виглядають як одне ціле. Співпроцесор виконує тільки свої специфічні функції (плаваючої арифметики, тригонометричні, експоненційні, логарифмічні), а всі інші – центральний процесор. Співпроцесор може працювати в реальному і захищеному режимах, а також перемикати режим розрядності 16/32.

Програмна модель більшості співпроцесорів спирається на загальну стекову організацію. Основу моделі утворює регістровий стек з восьми 80-бітових регістрів даних R0 – R7, які називають арифметичними регістрами. В цих регістрах зберігають дійсні числа у формі представлення з плаваючою комою у розширеному форматі. Діапазон представлених чисел подано в останньому рядку таблиці на с.52.

Крім арифметичних регістрів співпроцесор містить: 16-розрядні регістри керування CW, стану SW, слова тегів TW та 48-розрядні регістри вказівника команд IP та вказівника операндів DP.

Поточна вершина стеку (номер відповідного регістра) зберігається в полі TOP регістра стану. Внутрішня шина 84-розрядна. З кожним регістром пов'язане двобітве поле тегів, що дає змогу швидко аналізувати стан регістра.

Слово стану відображує загальний стан співпроцесора. Найстарший біт В вказує на зайнятість виконанням операції або наявність необслугованого запиту на переривання винятку. Біти C[0-3] визначають код умови, який формується при виконанні команди. Біт ES[7] встановлюється у випадку немаскованого переривання, а біти з 0 до 6 встановлюються при виникненні відповідних винятків.

У регістрі керування відображають опції виконуваних операцій. Тут задають маски винятків недійсних операцій, ділення на нуль, переповнення, антипереповнення, заокруглення та денормалізації. Поле PC задає точність (24, 53 або 64 біти), а поле RS визначає спосіб заокруглення (00 – до найближчого значення, 01 – за напрямком до “-∞”, 10 – за напрямком до “+∞”, 11 – за напрямком до нуля).

Регістри вказівника команд і даних зберігають фізичні адреси коду операції, яка виконується, і операнда.

Співпроцесори різних моделей відрізняються виконанням команди WAIT, роботою з префіксами команд, маскуванням винятків, представленням безмежності та форматами образів регістрів у реальному та захищеному режимах. Крім того, існують співпроцесори інших фірм (Cyrix, Weitek), які відрізняються потужністю, вимагають спеціального гнізда та програмного забезпечення.

### Контрольні запитання

1. Яка функція регістра ознак?
2. Як використовуються регістри загального призначення для організації виконання команд?
3. Що таке переривання і з якою метою їх застосовують?
4. Яка відмінність між внутрішніми і зовнішніми перериваннями?
5. Які переривання називають маскованими, а які немаскованими?
6. Який алгоритм опрацювання переривань?
7. Що називаємо виконавчою адресою?
8. Скільки і які компоненти використовують для формування виконавчої адреси?
9. Яка довжина сегмента у 16-розрядному процесорі?
10. Назвіть чотири типи сегментів.
11. Скільки режимів адресування є в 16-розрядному процесорі?
12. Що називаємо портом введення/виведення?
13. Скільки 8-бітових портів може мати 16-розрядний процесор?
14. Яку функцію виконує співпроцесор?
15. Які відмінності між процесорами 8086 і 80286?
16. Які головні відмінності між 32- та 16-розрядними мікропроцесорами?
17. За якими адресами (молодшими чи старшими) зберігається байт молодшого порядку у подвійному слові?
18. Скільки є типів сегментів?
19. Чи підтримує процесор одночасну роботу зі сторінками і сегментами?
20. Скільки 8-бітових портів може мати 32-розрядний процесор?

21. Яка відмінність між перериванням і винятком?
22. Як називають виняток, який виявляється й опрацьовується відразу після виконання команди з помилкою?
23. Який регістр зберігається у стеку перед опрацюванням переривання?
24. З якою метою переривання поділяють за пріоритетами?
25. Скільки і які компоненти використовують для формування логічної адреси?
26. Які типи адресних просторів Ви знаєте?
27. Що таке дескриптор?
28. Який біт дескриптора і в який стан (0 чи 1) його потрібно задати, щоб дескриптор був системним?
29. Для чого слугують біти D і P дескриптора?
30. Який розмір сторінки і сегмента 32-розрядного процесора?
31. Скільки рівнів захисту має 32-розрядний процесор?
32. Як можна змінити рівень привілеїв?
33. Як працює механізм посторінкової організації пам'яті?
34. Як влаштована кеш-пам'ять процесора?
35. Як працює алгоритм псевдо LRU?
36. Що означає віртуальний 8086-режим роботи 32-розрядного процесора?
37. З якою метою у процесорі запроваджено режим системного керування?
38. Які головні відмінності між цілочисловим процесором і спів-процесором?

### **2.1.2. Апаратна реалізація головних вузлів ПЕОМ**

#### **Системна плата**

Більшість ПЕОМ (за винятком переносних) конструктивно складаються з чотирьох обов'язкових складових – системного блока, клавіатури, миші та дисплея, які можуть доповнюватися додатковими пристроями – пристроями друку, сканування графічних образів, архівування інформації, модемами і т. ін. На рис. 2.25 подано фотографії сучасних ПЕОМ.



Рис. 2.25. Стационарні і переносна ПЕОМ

У системному блоці можна виділити як обов'язкові елементи (процесор, оперативна пам'ять, BIOS, магнітний диск, системна плата, блок живлення), так і частини, які встановлюються при потребі (контролер локальної мережі, внутрішній модем або факс-модем, швидкісний порт типу SCSI). На рис. 2.26 подана схема розташування складових системного блоку.

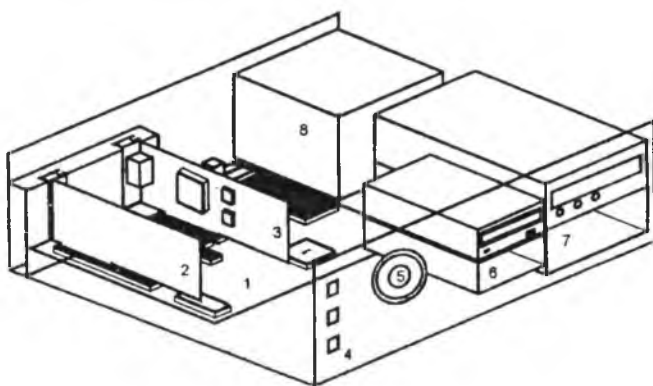


Рис. 2.26. Схема розташування складових системного блоку:

1 – системна плата, 2 – плата розширення ISA, 3 – плата розширення PCI, 4 – кнопка Reset та світлодіоди Power (індикація живлення) HDD (індикація звертань до ЖМД), 5 – гучномовець, 6 – відсік для 3,5" пристроїв (ГМД, ЖМД), 7 – відсік для 5,25" пристроїв (CD-ROM, mobile rack), 8 – блок живлення

Таке поєднання обов'язкової та варіативних частин дозволяє доволі просто нарощувати можливості системи шляхом встановлення або заміни деяких складових ПЕОМ. На рис. 2.27 подані зображення сучасних корпусів системних блоків для IBM-подібних ПЕОМ.





Рис. 2.27. Сучасні корпуси системних блоків

Основною частиною системного блоку є системна плата (System Board, Mother Board, Mainboard), на якій розміщені складові, що визначають архітектуру ПЕОМ:

- процесор;
- синхрогенератор, який забезпечує синхронізацію процесора й інших мікросхем;
- пам'ять (оперативна, постійна, CMOS);
- системна та локальні шини;
- обов'язкові системні засоби вводу/виводу.

Сучасні системні плати будуються з використанням кількох (1-2) мікросхем системної логіки (ChipSet). На рис. 2.28 подана структурна схема класичного двокристального набору мікросхем системної логіки для побудови системних плат та його реалізація у наборі Intel Springdall.

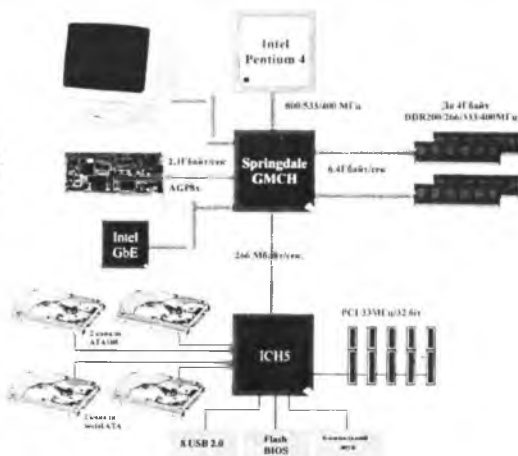


Рис. 2.28. Структурна схема класичного двокристального набору мікросхем системної логіки та його реалізація

Швидкі пристрої – контролер системної шини, контролер шини пам'яті, контролер графічної шини розміщують в одній мікросхемі (північний

міст), більш повільні – контролер IDE, USB, PCI, цифровий звуковий канал, вбудований контролер локальної мережі, інтерфейси для клавіатури, мишки, пристрою друку – в іншій мікросхемі (південний міст). Від характеристик набору мікросхем системної логіки істотно залежать параметри ПЕОМ. Габаритні розміри системних плат стандартизовані, що забезпечує ремонтпридатність і модернізованість системних блоків. Живлення на плату подається від блока живлення. Одна з точок кріплення плати забезпечує з'єднання лінії “земля” з металевим шасі корпуса, заземленого через мережевий шнур живлення. На рис. 2.29 подана схема розташування компонентів сучасної системної плати.

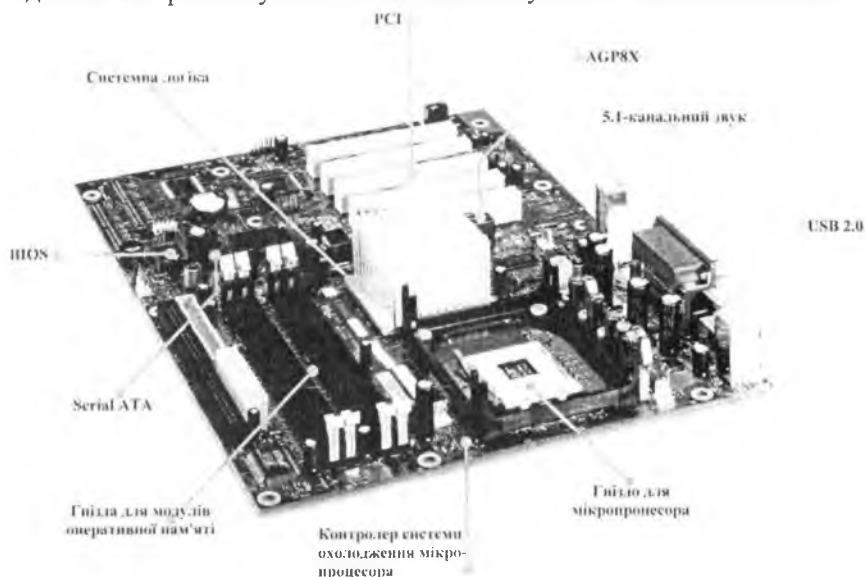


Рис. 2.29. Схема розташування основних компонентів системної плати

### 2.1.2.1. Процесор, співпроцесор

Процесор є основною функціональною складовою системного блока. Процесор призначений для опрацювання інформації за програмами, які розміщують в оперативній пам'яті. Від його характеристик (швидкодії, розрядності регістрів та шин процесора) істотно залежать споживачькі якості ПЕОМ. Основна частина команд процесора орієнтована на операції з цілими числами (у двійковому коді). Для виконання операцій над дробовими числами використовували два підходи:

- програмна реалізація десяткової арифметики (зменшується швидкодія);
- апаратна реалізація десяткової арифметики (співпроцесор “плаваючої арифметики”).

Використання співпроцесора “плаваючої арифметики” дозволяє підвищити швидкодію ПЕОМ при опрацюванні числової інформації. Для мікропроцесорів I8086, I8088, I80286, I80386SX, I80486SX використовували зовнішній мікропроцесор (I8087, I80287). Починаючи з I80386DX співпроцесор “плаваючої арифметики” розташовується в одному корпусі із процесором, що дозволяє підвищити швидкість обміну інформацією між процесором і співпроцесором за рахунок використання внутрішньої шини мікропроцесора.

У таблиці 2.5 подані основні характеристики процесорів Intel.

Табл. 2.5. Основні характеристики процесорів Intel

Процесор	Розрядність (біт)			Внутрішній кеш (байт)	Вбудований співпроцесор	Внутрішня тактова частота (МГц)
	Внутрішніх регістрів	Шини даних	Шини адреси			
I8086	16	16	20	-	-	4.77-8
I8088	16	8	20	-	-	4.77-8
I80286	16	16	24	-	-	6-25
I80386SX	32	16	24	-	-	16-33
I80386DX	32	32	32	-	+	25-40
I80486SX	32	32	32	8K	-	16-33
I80486DX	32	32	32	8K	+	25-50
I80486DX2	32	32	32	16K	+	40-80
I80486DX4	32	32	32	16K	+	75-120
Pentium I	32	64	32	2 x 8K	+	60-133
Pentium II	64	64	32	2 x 16K	+	233-450
Pentium III	64	64	32	256K	+	450-1000
Pentium IV	64	64	32	256K – 2M	+	1000-3800

З часу створення 16-бітного мікропроцесора I8086 мікропроцесори для IBM-подібних ПЕОМ пройшли довгий шлях розвитку. Наступний матеріал коротко ілюструє розвиток мікропроцесорів для IBM-подібних ПЕОМ.

### 2.1.2.2 Мікропроцесор Intel 8086

Свого часу поява 16-розрядного МП Intel 8086(88) визначила цілий напрямок у розвитку і застосуванні мікропроцесорних комплексів (МПК). У колишньому СРСР було також розроблено і виготовлено мікропроцесорний комплект К1810ВМ86, який був функціонально тождним Intel 8086 і був однокристальним (кремнієвий кристал) 16-розрядним МП зі швидкістю ~2,5 млн операцій за секунду. Система адресації дозволяла адресувати 1 Мб оперативної пам'яті. Система команд нараховувала 135 команд.

У комплекті містився також математичний співпроцесор Intel 8087 чи К1810ВМ87 — однокристальний 16-розрядний процесор для виконання команд арифметики з плаваючою комою.

Водночас з МП 8086 створено МП 8088 – однокристальний 16-розрядний МП з більшою швидкодією, ніж 8080 (~ у 5 разів), однак з 8-розрядною зовнішньою магістраллю для зв'язку з пристроями введення/виведення. Набір регістрів і система команд така ж, як і у 8086. Завдяки більшій розрядності 8086 працює на 20-60 % швидше, ніж 8088 з однаковою тактовою частотою.

Крім цього, у комплект входять: генератор тактових сигналів, системний контролер, арбітр шини та ін.

Головні властивості цих МП:

- розвинена регістрова структура, яка значно зменшує кількість звертань до пам'яті;
- конвеєрний принцип виконання команд з попереднім вибиранням, що забезпечує максимальну пропускну здатність системної магістралі;
- наявність мультиплексованої шини адреси / даних;
- багатфункціональне використання виводів МП;
- здатність до координації взаємодії кількох процесорів (зокрема співпроцесорів).

Характерною рисою цих процесорів, на відміну від 4- та 8-розрядних, є те, що вони зорієнтовані на паралельне виконання команд і вибирання з пам'яті. Тобто є дві частини, що працюють асинхронно: спряження з магістраллю й опрацювання даних чи команд.

На базі аналога Intel 8086 – мікропроцесорі КМ1810ВМ86 були побудовані такі ПЕОМ: ЕС1840, ЕС1841, “Искра 1030”, Нейрон, “Поиск-2”, ЕС7978, “Поиск-1”, Практик (К1810ВМ88). Структурна схема мікропроцесора Intel 8086 подана на рис. 2.30.

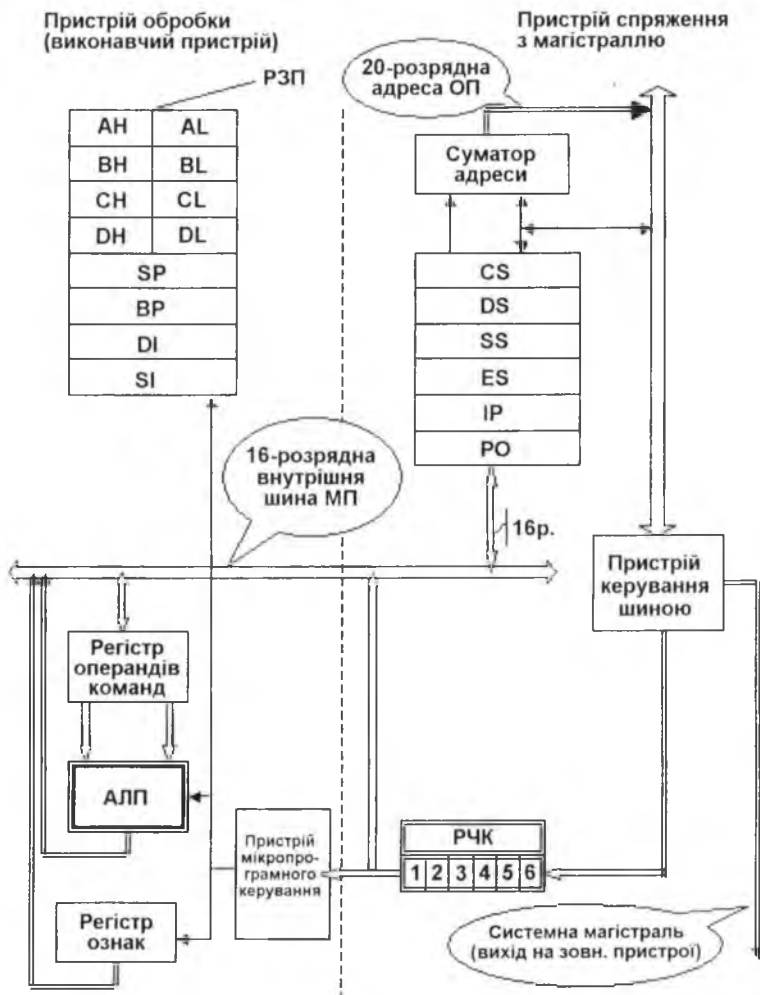


Рис. 2.30. Мікропроцесор Intel 8086

У складі пристрою спряження з магістраллю (ПСМ) крім регістрових структур є 16-розрядний суматор адреси (СМА) та пристрій керування шиною. Цикл вибирання з пам'яті слова розпочинається тоді, коли у черзі вивільняються принаймі 2 байти. Виконання команд відбувається у логічній послідовності, яка задана програмою, оскільки в черзі є ті команди і в тій послідовності, як вони були у комірках пам'яті. Якщо

трапляється команда передачі керування іншій комірці пам'яті, то пристрій спряження очищує регістр черги команд, вибирає команду за адресою переходу, передає її у пристрій керування і розпочинає нове заповнення регістрів. У разі повернення з програми (після CALL) чи внаслідок переривання відновлюється черга команд, адреси яких обчислюються за допомогою СМА. Якщо процесор виконує цикл читання або записування, то на час дії циклу вибирання команд припиняється.

20-розрядну фізичну адресу формує СМА шляхом зсуву базової адреси сегмента на чотири розряди вліво і додавання до нього 16-розрядної адреси зміщення. Адреса зміщення є в одному з регістрів вказівників або індексних і дає змогу адресувати 64 Кб вибраного сегмента; 20-розрядна адреса дає змогу працювати з пам'яттю 1 Мб.

Якщо отримане у СМА значення перевищує значення найстаршої адреси сегмента, то фізичну адресу знову відраховують від початку сегмента.

**Пристрій опрацювання даних** призначений для виконання операцій з опрацювання даних, і він складається з:

- мікропрограмного пристрою керування;
- 16-розрядного арифметико-логічного пристрою (АЛП);
- восьми 16-розрядних регістрів загального призначення;
- регістра ознак.

Команди, вибрані пристроєм спряження з пам'яті через регістр черги команд (РЧК), надходять у пристрій мікропрограмного керування. Цей пристрій містить пам'ять мікрокоманд, декодує команди і виробляє послідовність мікрокоманд, що керує процесом опрацювання.

В АЛП виконуються арифметичні і логічні операції над 8- і 16-розрядними числами з фіксованою комою. Для прискорення внутрішніх пересилань даних всі регістри і магістралі мають 16 розрядів. Прямого зв'язку ПО із зовнішньою магістраллю немає, а обмін даними з ПСМ відбувається через регістр обміну (РО).

### 2.1.2.3 Мікропроцесор 180286

Цим процесором представлено друге покоління 16-розрядних процесорів (випуск розпочато у 1982 р.). Найсуттєвіша відмінність від 8086 – це наявність нового механізму керування адресуванням пам'яті. Має спеціальні засоби для підтримання механізму перемикання задач. Розширена система команд. МП працює у двох режимах:

- *8086 Real Address Mode* – режим реального адресування, повністю сумісний з 8086 (1 Мб);
- *Protected Virtual Address Mode* – захищений режим віртуального адресування (16 Мб). Використання сторінкового механізму адресування дає змогу заадресувати 1 Гб віртуальної пам'яті кожної задачі.

Призначення регістрів головню збігаються з 8086. Різниця є у регістрі ознак: біти з 0 до 11 збігаються з 8086, а додано у 80286: (12,13) IOPL (Input Output Privilege Level) - рівень привілейованості введення/виведення, (14) NT (Nested Task Flag) – ознака вкладення задач. Режим роботи процесора регламентують біти регістра стану машини (MSW): PE – переводить процесор у захищений режим, MP – відповідає за моніторинг співпроцесора, EM – дозволяє програмну емуляцію співпроцесора, TS – встановлює режим перемикання задач.

Наприклад, сукупність ознак MP=0, EM=0, TS=0, яка встановлюється при скиданні процесора в нуль, забезпечує повну сумісність з процесором 8086(88). Комбінація ознак MP=1, EM=0 відповідає наявності співпроцесора на платі, комбінація MP=0, EM=1 встановлює режим програмної емуляції співпроцесора.

Збільшено тактову частоту. Зміни в архітектурі дозволили скоротити час виконання операцій: процесор 80286 з тактовою частотою 12,5 МГц працює у 6 разів швидше, ніж 8086 з тактовою частотою 4,77 МГц.

Під керуванням MS DOS 80286 переважно використовували у реальному режимі, тобто він працює як швидкий 8086. Захищений режим використовує ОС типу XENIX, UNIX, OS/2, Netware 286 і MS Windows.

З'являються поняття сторінки, селектора та дескриптора сегментів. Усі ці поняття стосуються нового режиму адресування пам'яті. Процесор 80286 має засоби контролю за переходом через межу сегмента, причому цей контроль здійснюється і в реальному режимі. У порівнянні з 8086 є відмінності у визначенні сегментів:

- базова адреса сегмента має розрядність 24 біти, що дозволило адресацію 16 Мб фізичної пам'яті;
- сегментні регістри CS, DS, SS, ES зберігають не самі базові адреси сегментів, а селектори, які визначають у спеціальних таблицях оперативної пам'яті дескриптори сегментів, де міститься базова адреса, розмір сегмента та його атрибути.



Рис. 2.31. Мікропроцесор Intel 80286

Апаратні і програмні переривання працюють так само, як і у 8086. Кожному номеру переривання відповідає елемент у таблиці дескрипторів переривань (IDT). У реальному режимі таблиця переривань організована так само, як і у 8086 і розташована у пам'яті з молодшими адресами. Команда LIDT дає можливість змінити її положення та розмір з 03FFh до 007Fh. У захищеному режимі таблиця IDT містить від 32 до 256 8-байтових дескрипторів переривань і може розташовуватися у будь-якому місці фізичної пам'яті.

Співпроцесор 80287 працює у реальному і захищеному режимах. Апаратний інтерфейс розрахований на приєднання до локальної шини процесора 80286 та 80386. Регістри даних R0-R7 – 80-розрядні, а вказівники команд залежні від розрядності операндів (можливі і 32-бітні форми).



#### 2.1.2.4 Мікропроцесор I80386

Серію 32-розрядних процесорів фірми Intel започаткував мікропроцесор Intel 80386 (випущений у 1985 р.). Ці процесори увібрали в себе всі властивості попередників, тобто 16-розрядних процесорів і забезпечили програмну сумісність зі значним доробком програмного забезпечення, написаного для них. Процесори забезпечують систему захисту оперативної пам'яті й уведення/виведення, перемикання задач. Вони мають розширену систему команд, яка також включає всі команди 8086 та 80286. Можливо, найціннішим є те, що у цих процесорах значно відсунуто верхню межу адресування оперативної пам'яті. Процесори можуть працювати у двох режимах: реального та захищеного віртуального адресування.

У реальному режимі процесор працює як дуже швидкий 8086, проте з 32-розрядним розширенням. Реальний режим також використовують для підготовки мікропроцесора до роботи в захищеному режимі.

Віртуальний режим забезпечує доступ до дуже складного, сучасного способу керування пам'яттю, а також для підкачування (заміщення чи свопінгу) сторінок та інших можливостей мікропроцесора.

У віртуальному режимі програмне забезпечення процесора може розв'язувати задачі за ступенем складності такі ж, як у 8086 і 80286. Цей режим дає змогу виконувати програмне забезпечення та прикладні програми 8086 одночасно з операційною системою та прикладним забезпеченням 32-розрядного процесора.

Віртуальні задачі 32-розрядних МП можуть бути захищені одна від іншої та операційної системи шляхом використання заміщення сторінок, смуляції команд уведення/виведення.

Для суміщення комп'ютера з високоефективними системами інтерфейс шини 32-розрядних МП має конвеєрне опрацювання даних, динамічне резервування шини даних і прямі сигнали BYTE ENABLE.

Головною особливістю є апаратна реалізація багатовисхідного програмного середовища, яке забезпечує сумісну роботу програм користувачів, зорієнтованих на операційні системи UNIX, DOS і APX86.

Ознайомимося з архітектурою цих процесорів детальніше.

Мікропроцесор Intel80386 складається з центрального цілочисельного процесора, блока керування пам'яттю та шини інтерфейсу (рис. 2.32).

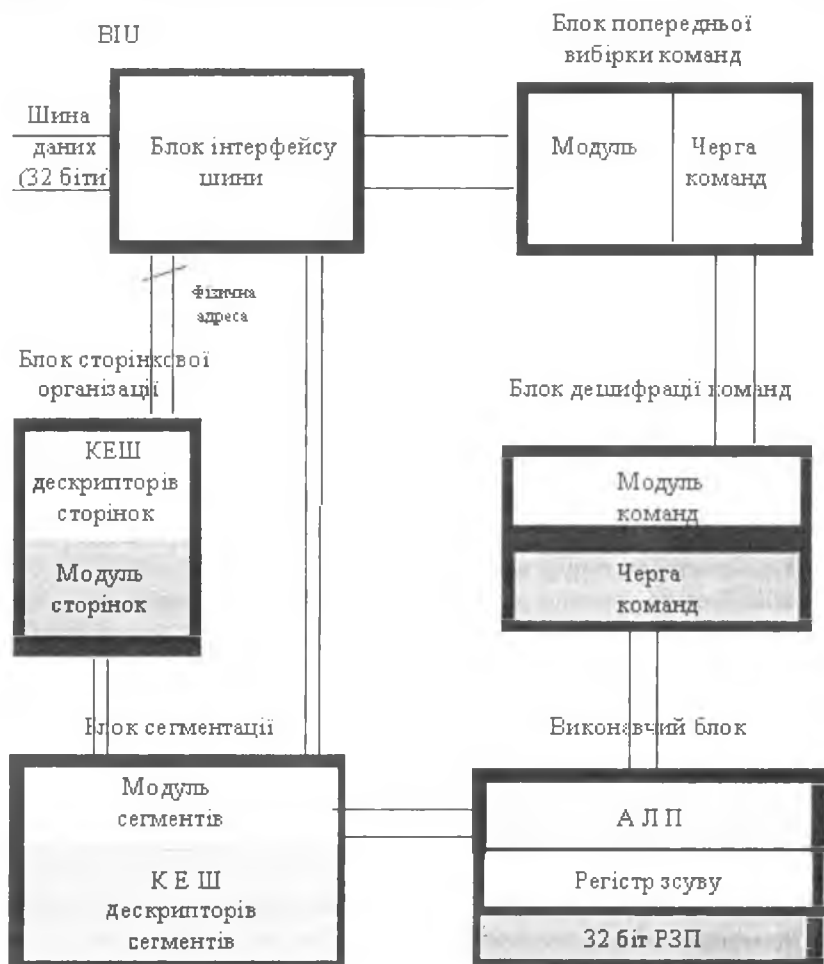


Рис. 2.32. Структурна схема мікропроцесора Intel 80386

Центральний процесор має виконавчий блок, що містить вісім 32-розрядних регістрів загального призначення, які використовують як для обчислення адреси, так і для операцій з даними, та арифметико-логічний пристрій.

Пристрій опрацювання команд дешифрує коди операцій команд і поміщає їх у чергу для невідкладного виконання виконавчим блоком. Внутрішня черга команд має розмір 16 байт.

Пристрій керування пам'яттю складається із блока сегментації та пристрою заміщення сторінок (підкачування).

Сегментація дає змогу керувати простором логічної адреси.

Механізм заміщення сторінок працює "вниз" і паралельний процесові сегментації, що дає змогу керувати простором фізичної адреси. Одна сторінка його становить 4 Кб.

Для використання системи віртуальної пам'яті 80386 підтримує повну відновлюваність для всіх помилок на сторінці і сегменті. Пам'ять може бути організована в один або більше сегментів різної довжини до 4 Гб.

Будь-яке завдання на 80386 може мати 16 381 сегмент, довжиною до 4 Гб кожний, тобто забезпечує 64 Тб віртуальної пам'яті для кожної задачі.

Блок сегментації забезпечує чотири рівні захисту для ізоляції та захисту пристроїв і операційної системи одне від одного. Конструкція мікропроцесора дає змогу всі системи сполучати в одне ціле.

Передбачена можливість використання математичного співпроцесора 80387, програмно сумісного з 8087 і 80287. Порівняно з попередніми моделями збільшено продуктивність обчислень, додано кілька команд, запроваджено режим енергозбереження та ін.

Процесор I80386 з двобайтною шиною даних та 24-розрядною адресною шиною позначався I80386SX, чотирибайтною шиною даних та 32-розрядною адресною шиною – I80386DX.

### 2.1.2.5 Мікропроцесор I80486

Архітектура МП Intel 80486 дуже подібна до архітектури Intel 80386. Однак вирізняє її дві суттєві деталі – організація внутрішньої кеш-пам'яті першого рівня та наявність пристрою з плаваючою комою (співпроцесора) на кристалі.

Тут дані й адреси передаються по внутрішніх 32-розрядних двоскерованих шинах, а для покращення обміну даними з кешом запроваджено режим пакетного передавання даних (Burst Mode). Запроваджено елементи RISC архітектури, що дало змогу найуживаніші команди виконувати протягом одного такту. Крім того:

- додано нову ознаку AC (Alignment Check) контролю вирівнювання (біт I8);

- додано нові біти у регістрах керування CRO (NE, WP, AM, CD) та CR3 (PCD, PWT). Біт CD (CacheDisable) – керування кешом, біт NW (NotWrite-through) – наскрізним записом у зовнішню пам'ять;

- додано нові команди (три прикладні – BSWAP (Byte SWAP) обміну байтами, XADD (eXchange and ADD) обміну і додавання, CMPXCHG (CoMPare and eXCHanGe) порівняння й обміну) і три системні команди для керування кеш-пам'яттю і буфером TLB);

- уведено функції контролю вирівнювання операндів;
- чергу команд збільшено до 16;
- розширено засоби тестування (регістри TR3, TR4, TR5 для тестування внутрішнього кешу);
- застосовано множення тактової частоти системної плати (внутрішня частота DX2 дорівнює подвійній зовнішній, а у DX4 – кратність може бути 2, 2,5 і 3).

Спрощена внутрішня архітектура процесора 80486 показана на рис. 2.33.

8 Кбайтова кеш-пам'ять команд і даних використовує дві 32-бітові шини даних спільно з пристроєм сегментації, цілочисловим пристроєм та пристроєм з плаваючою комою. Ці дві шини можна використати сумісно як 64-бітову шину для передавання між пристроями. У цьому випадку 64-бітові дескриптори сегментів передаються з кеш-пам'яті у пристрій сегментації, 32 біти прямо передаються по одній шині даних, а інші 32 – через цілочисловий пристрій, отже, всі 64 біти досягають пристрою сегментації одночасно.

У МП 80486, як і у 80386, застосовано конвеєризацію команд: одночасно виконуються операції, пов'язані з попереднім вибиранням команд, дешифруванням команд, виконанням мікрокоду, а також цілочислові операції, операції з плаваючою комою, сегментації, сторінкового перетворення, керування кеш-пам'яттю і шинним інтерфейсом.

Пристрій попереднього вибирання команд оперує 16-байтними блоками команд, які одночасно подаються і в кеш-пам'ять. Якщо у виконуваній програмі зустрічаються цикли, то пристрій попереднього вибирання отримує копії команд, що виконувалися раніше, з кеш-пам'яті, тобто звернення до основної пам'яті не відбувається.

Пристрій з плаваючою комою має такий самий набір регістрів і систему команд, як і співпроцесор 80387, однак працює в 3-4 рази швидше.



Рис. 2.33. Мікропроцесор Intel 80486

### 2.1.2.6 Мікропроцесор Pentium I

Процесори Pentium фірми Intel належать до п'ятого покоління процесорів x86. Вони сумісні з попередніми 32-розрядними МП, однак мають 64-бітову шину даних. Однією з головних відмінностей є його суперскалярна архітектура.

*Скалярним* називають процесор з єдиним конвєрсом виконання команд (усі процесори Intel до 80486 включно).

Суперскалярний процесор має більше одного конвеєра (Pentium має 2, Pentium Pro – 3), які здатні опрацьовувати інструкції паралельно.

Інші відмінності цього процесора такі:

- застосування технології динамічного передбачення розгалужень у програмах разом з виділеним внутрішнім кешом команд обсягом 8 Кб, що забезпечує максимальне завантаження конвеєрів;

- внутрішній (рівень 1) кеш даних обсягом 8 Кб, на відміну від 486-го, працює з відкладеним (до вивільнення зовнішньої шини) записом і налаштовується на режим наскрізного чи зворотного записування, підтримуючи протокол забезпечення когерентності пам'яті MESI (Modified-Exclusive-Shared-Invalid);

- зовнішня шина даних для підвищення продуктивності має розрядність 64 біти, що потребує відповідної організації пам'яті;

- вмонтований співпроцесор за рахунок архітектурних поліпшень (конвеєризації) у 2-10 разів перевершує продуктивність FPU-486;

- застосовано виявлення помилок внутрішніх пристроїв (внутрішній контроль паритету) і зовнішнього інтерфейсу шини, контролюється паритет шини адреси;

- уведена можливість побудови функціонально надлишкової дво-процесорної системи;

- реалізовано інтерфейс побудови двопроцесорних систем з симетричною архітектурою (починаючи з другого покоління Pentium);

- уведені засоби керування енергопостачанням;

- застосована конвеєрна адресація шинних циклів;

- скорочений час (кількість тактів) виконання інструкцій;

- уведено трасування інструкцій і моніторинг продуктивності;

- розширені можливості віртуального режиму – уведено віртуалізацію ознаки переривань;

- уведена можливість оперування сторінками розміром 4 Мб у режимі переадресування (Paging).

Структурна схема процесора зображена на рис. 2.34.

Процесори Pentium першого покоління (P5) з тактовою частотою 60 і 66 МГц живилися від джерела напруги 5 В, що призводило до великого тепловиділення (за частоти 66 МГц – 16 Вт). Їх випускали в корпусах PGA-273 (матриця 21х21), для встановлення цих процесорів призначене спеціальне роз'язтя (Socket 4).



Рис. 2.34. Структурна схема процесора Pentium

Перші моделі процесорів мали помилку в FPU, яка полягала у втраті точності під час виконання ділення з деякими поєднаннями операндів. Помилка могла з'являтися від 4-го до 19-го розряду після десяткової коми. З початку 1995 р. процесори почали випускати вже без помилок.

Pentium Overdrive 120 і 133 МГц – варіант процесора Pentium другого покоління (зі зниженою енергоємністю і подвоєнням частоти) – призначені для заміни процесорів Pentium першого покоління. Вони мають той самий корпус PGA-273.

Процесори Pentium другого покоління (P54) живляться від джерела напруги 3,3 В і менше, що суттєво знижує розсіювану потужність. Їхні входні і вихідні сигнали зіставні з ТТЛ-логікою, однак для входу допустимий рівень сигналу обмежений значенням 3,3 В (крім тактових входів CLC і PICCLC, які допускають рівень 5 В). Сучасніші моделі

використовують технологію зниження напруги VRT (Voltage Reduction Technology). Напруга VCC для інтерфейсних схем і надалі дорівнює 3,3 В, а для живлення ядра, яке потребує близько 90% потужності, VCC знижено до 2,9 В, що знижує розсіювану потужність.

Процесори виготовляли у корпусах SPGA-296 з шаховим розташуванням виводів, для їхнього встановлення передбачені сокети 5 і 7. Сокет 7 має дві шини живлення: VCC2 для живлення ядра процесора і VCC3 для живлення інтерфейсних схем.

У процесорах другого покоління застосовано внутрішнє помноження частоти: інтерфейсні схеми зовнішньої системної шини працюють на частотах 50, 60 або 66,66 МГц, а ядро процесора – на вищій частоті (75, 90, 100, 120, 133, 150, 166, 180 і 200 МГц). Поділ частот дає змогу реалізувати досягнення технології виготовлення процесорів, які суттєво випереджають можливості підвищення продуктивності пам'яті та інших традиційних компонентів комп'ютера. Коефіцієнт множення (1,5; 2,0; 2,5 чи 3,0) задано комбінацією рівнів сигналів на входах BF0, BF1 у межах, які дозволені специфікацією тактової частоти процесора. Незалежність задання зовнішньої частоти і коефіцієнта множення дає змогу одну і ту ж внутрішню частоту задавати різними способами. Наприклад, 100 МГц можна отримати як  $50 \cdot 2$  і як  $66,66 \cdot 1,5$ . У загальному випадку ліпше застосовувати останній варіант, оскільки при цьому шина PCI (Peripheral Component Interconnect bus) буде працювати на частоті 33, а не 25 МГц.

### 2.1.2.7 Мікропроцесори PentiumPro і Pentium II

Процесор PentiumPro прийнято вважати за початок процесорів шостого покоління. З погляду виконання команд це покоління вирізняє наявність динамічного виконання, що передбачає виконання інструкцій всередині процесора не у тому порядку, який визначено програмою. Фактично це поєднання методів передбачення розгалужень, аналізу проходження даних та віртуального виконання. Такий підхід збільшує продуктивність не за рахунок збільшення тактової частоти, а внаслідок поліпшення архітектури процесора. Окрім цього, запроваджено ще низку вдосконалень:

- безпосередньо у корпус процесора інтегровано синхронний кеш другого рівня (L2-кеш), який має обсяг 256 Кб, 512 Кб, 1 Мб і приєднаний до внутрішньої шини;



- застосовано архітектуру подвійної незалежної шини. Системна шина призначена для спілкування ядра з основною пам'яттю та інтерфейсними пристроями, а інша – для обміну із вторинним кешом;
- система команд розширена інструкціями пересилання даних, що дає змогу зменшити кількість умовних переходів;
- поліпшено контроль за роботою шини та кешу;
- архітектура процесора та інтерфейс системної шини дають змогу об'єднати до чотирьох процесорів на одній шині у симетричну мультипроцесорну систему.

Наступна архітектурна реалізація фірми Intel – процесор Pentium II. Це поєднання архітектури процесора PentiumPro та технології MMX (MultiMedia eXtention), тобто розширення системи команд МП для роботи з мультимедійними додатками. Однак Pentium II має і свої особливості. Зокрема, для його виготовлення застосовано нову технологію корпусів – так званий картридж з друкованим крайовим роз'ємом, куди виведено системну шину. На цьому картриджі змонтовано власне ядро процесора і вторинний кеш ємністю 512 Кб. Первинний кеш порівняно з PentiumPro подвоєно і він становить 16 Кб даних і 16 Кб команд.

Системна шина забезпечує приєднання двох процесорів. Шина фізичної адреси розширена до 36 біт, що дає змогу заадресувати 64 Гб фізичної пам'яті.

## Про розширення MMX

Розширення MMX застосовують для організації мультимедійної роботи та опрацювання 2D і 3D-графічних файлів. Головна його ідея полягає у використанні технології ОКБД (див. 1.4).

У систему команд уведено 57 нових команд для одночасного опрацювання кількох одиниць даних. Для роботи застосовано нові типи упакування 64-бітових даних, а саме:

- упаковані байти (вісім байт);
- упаковані слова (чотири слова);
- упаковані подвійні слова (два подвійних слова);
- почотверенне слово (одне слово).

Ці типи даних можна опрацьовувати у регістрах MMX0–MMX7, які є молодшими бітами стеку 80-бітових регістрів співпроцесора. Збіг

регістрів для опрацювання кодів MMX чи FPU є особливістю MMX розширення і потребує від програміста уваги у використанні цих регістрів. Часта зміна використання кодів MMX чи FPU може знизити продуктивність процесора внаслідок необхідності збереження і відновлення стану FPU.

Технологія MMX має ще одну особливість, яка полягає у підтримці команд *арифметики з насиченням*. Суть такої арифметики полягає у тому, що замість переповнення чи антипереповнення фіксується максимально чи мінімально можливе значення величини. Це дуже важливо, наприклад, під час роботи з графічними об'єктами, де відбувається обчислення кольору.

Додаткові команди можна розділити на такі групи:

- арифметичні (+, -, x, +x);
- логічні (I, I-HE, АБО, АБО-НЕ);
- порівняння;
- перетворення форматів;
- зсуви;
- пересилання даних;
- очищення MMX (у слові тегів).

Команди MMX доступні з будь-якого режиму процесора. Вони не породжують нових винятків і не впливають на стан регістра ознак.

### 2.1.2.8 Мікропроцесор Pentium III

У 1999 р. Intel Corp. оголосила про випуск Pentium III (Katmai).

Можна вважати, що новий процесор Pentium III є не що інше, як Pentium II з новими 70 командами (головно MMX). До нових команд належать так звані команди Streaming SIMD Extensios (SSE), призначення яких – прискорення операцій у програмах опрацювання 3D графіки. Швидше опрацьовується 2D графіка. Фірма Intel Corp. дала змогу програмістові керувати кешом. Тепер, якщо потрібно залишити у кеші навіть маловживані дані, вони не будуть автоматично заміщені новими.

Кожний чіп Pentium III повідомляє свій унікальний номер.

Невдовзі інші фірми почали виготовляти клонові процесори AMD-K6-III – 400 МГц та Athlon 800. Уперше у архітектурі запропоновано трирівневий кеш. У процесорі AMD-K6-III кеш організовано як

двопортову зі зворотним записом розділену кеш-пам'ять першого рівня L1 ємністю 64 Кб (по 32 Кб на команди і дані), другого рівня L2 ємністю 256 Кб, яка інтегрована в одному кристалі з ядром, і L3, яка розташована на системній платі. Процесор містить 21,3 млн транзисторів на кристалі площею 118 мм<sup>2</sup> і споживає струм 12,4 А з напругою живлення 2,4 В.

### 2.1.2.9 Мікропроцесор Pentium IV

У серпні 2000 р. оголошено про випуск процесора Pentium IV. Розробники намагались якнайліпше пристосувати процесор до його використання в мережі Інтернет, для опрацювання зображень, розпізнавання мови, відеопотоків, 3D графіки, мультимедіа та багатозадачних обчислювальних середовищ. Застосовано нову мікроархітектуру NetBurst, яка повинна забезпечити зростання продуктивності й тактової частоти на перспективу. Суть нової мікроархітектури NetBurst:

- застосовано технологію Hyper Pipelined (гіперконвеєр), тобто довжину конвеєра подвоєно до 20 етапів;
- подвоєно, порівняно з ядром процесора, тактову частоту роботи арифметико-логічних блоків. Це дозволяє процесору виконувати деякі команди за половину такту, а цілочислові операції виконувати з подвоєною швидкістю;
- удосконалено системну шину (400-533 МГц), яка втричі швидша від шини процесора Pentium III. Швидкість передавання даних між процесором Pentium IV і контролером пам'яті досягає 3,2 Гб/с;
- кеш-пам'ять процесора має 64-байтовий доступ (попередні процесори мали 32-байтовий доступ). У кеші першого рівня зберігають декодовані команди (приблизно 12 000 мікрокоманд), що помітно поліпшує швидкодію кешу і збільшує ефективність його використання. Pentium IV містить кеш другого рівня ємністю 256 Кб та швидкістю передавання даних 48 Гбіт/с;
- удосконалено систему динамічного виконання команд та покращено схеми передбачення розгалужень у програмі;
- додано 144 нових SIMD команди (128-розрядних), які дозволяють суттєво пришвидшити роботу різних програм.

На рис. 2.35 подана спрощена блок-схема мікропроцесора Intel Pentium IV.

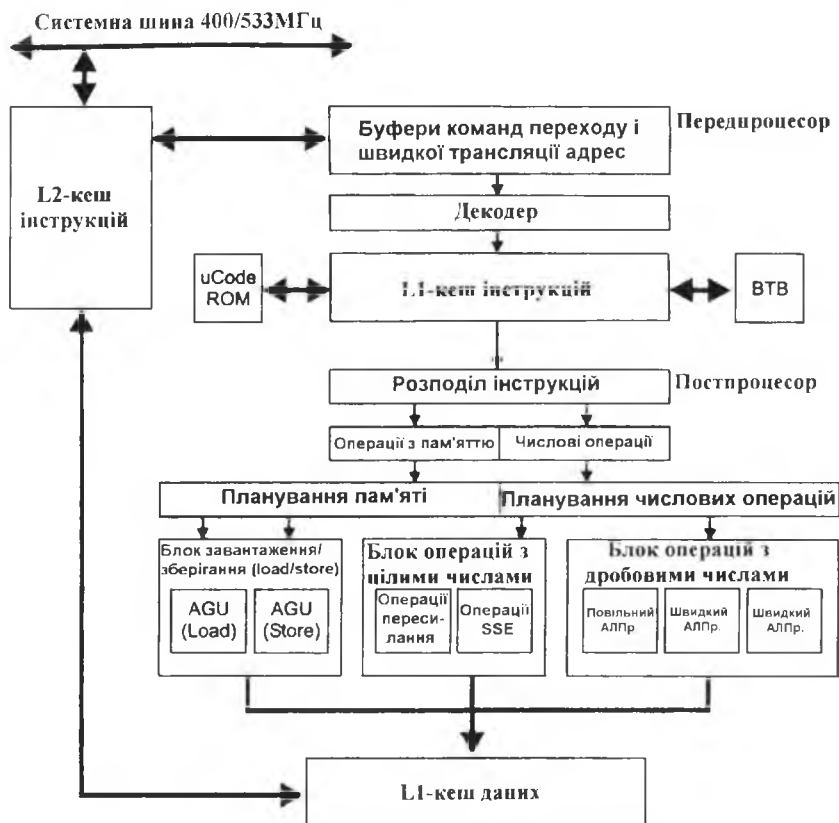


Рис. 2.35. Спрощена блок-схема мікропроцесора Intel Pentium IV (AGU – пристрій генерування адреси)

L2 – кеш другого рівня для зберігання частини команд програми, яка виконується у цей час, L1 – кеш першого рівня для зберігання інформації (даних).

У мікропроцесорі Intel Pentium IV (корпус LGA 775) 3.6 ГГц фірма Intel змінила форму контактів на корпусі мікропроцесора – металеві ніжки замінено на круглі контакти, а отвори у гнізді мікропроцесора на пружні контакти. На рис. 2.36 подано зображення мікропроцесора Intel Pentium IV LGA 775 і гнізда Socket T.

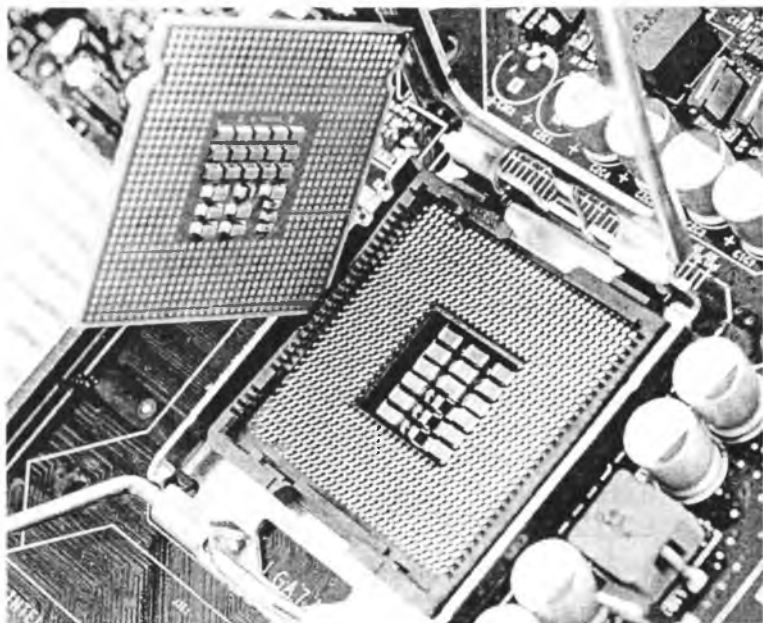


Рис. 2.36. Встановлення мікропроцесора Intel Pentium IV у гніздо Socket T

Табл. 2.6. Мікропроцесори фірми Intel

Intel Pentium IV			Intel Celeron		
Назва	Тактова частота	Гніздо	Назва	Тактова частота	Гніздо
Pentium 4 3,8 ГГц	3,8 ГГц	Socket T	Celeron 2,8 ГГц	2,8 ГГц	Socket 478
Pentium 4 3,6 ГГц	3,6 ГГц	Socket T	Celeron 2,6 ГГц	2,6 ГГц	Socket 478
Pentium 4 3,4 ГГц	3,4 ГГц	Socket T	Celeron 2,4 ГГц	2,4 ГГц	Socket 478
Pentium 4 3,06 ГГц	3,06 ГГц	Socket 478	Celeron 2,1 ГГц	2,1 ГГц	Socket 478
Pentium 4 2,8 ГГц	2,8 ГГц	Socket 478	Celeron 2,0 ГГц	2,0 ГГц	Socket 478
Pentium 4 2,66 ГГц	2,66 ГГц	Socket 478	Celeron 1,8 ГГц	1,8 ГГц	Socket 478
Pentium 4 2,6 ГГц	2,6 ГГц	Socket 478	Celeron 1,7 ГГц	1,7 ГГц	Socket 478
Pentium 4 2,53 ГГц	2,53 ГГц	Socket 478			
Pentium 4 2,5 ГГц	2,5 ГГц	Socket 478			
Pentium 4 2,4 ГГц	2,4 ГГц	Socket 478			
Pentium 4 2,26 ГГц	2,26 ГГц	Socket 478			
Pentium 4 2,2 ГГц	2,2 ГГц	Socket 478			
Pentium 4 2,0 ГГц	2 ГГц	Socket 478			
Pentium 4 1,9 ГГц	1,9 ГГц	Socket 478			
Pentium 4 1,8 ГГц	1,8 ГГц	Socket 478			

Слід зауважити, що паралельно із мікропроцесорами Pentium IV (тактова частота 3,8 – 1,8 ГГц, кеш 512Кб – 4Мб) фірма Intel продукує дешевші (у кілька разів!) мікропроцесори Celeron (тактова частота 2,8 – 1,7 ГГц, кеш 128-256Кб), які мають дещо нижчу тактову частоту, аніж найновіші Pentium IV та менший розмір вбудованого кешу L1, L2. Значення тактових частот і розміри вбудованого кешу подані на момент написання книги.

### 2.1.2.10. Intel-сумісні процесори інших виробників

Крім мікропроцесорів фірми Intel у IBM-подібних ПЕОМ використовували мікропроцесори фірм IBM, Texas Instruments, AMD (K6-2, Duron, Athlon), Via (Cyrix). Сучасні мікропроцесори є надзвичайно складними пристроями. Вивчати їх будову на детальному рівні потрібно напевно лише розробникам мікропроцесорів (та й ця інформація є комерційною таємницею). Разом з тим загальні принципи побудови сучасних мікропроцесорів відомі.

На рис. 2.37-2.38 подана спрощені блок-схеми мікропроцесорів AMD Athlon XP, Via, а на рис. 2.39 – 2.40 – фотографії мікропроцесорів.

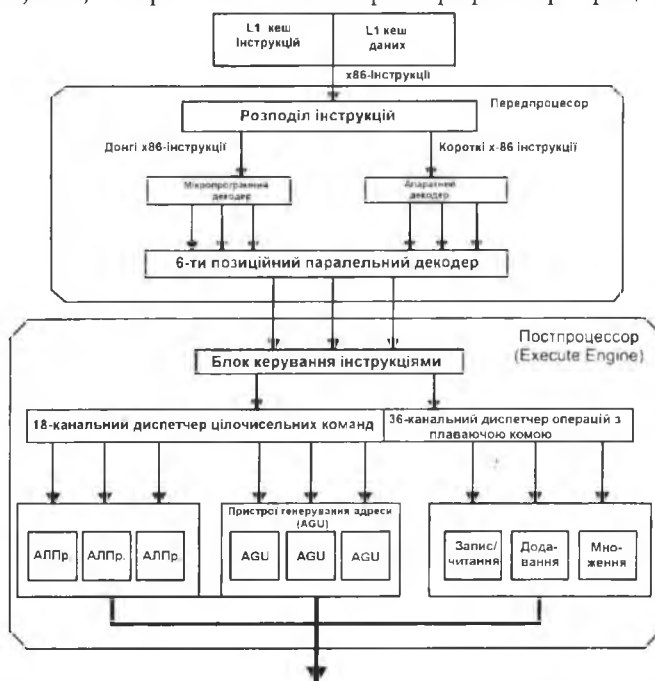


Рис. 2.37. Спрощена блок-схема мікропроцесора AMD Athlon XP

L2 – кеш другого рівня для зберігання частини команд програми, яка виконується у цей час, L1 – кеш першого рівня для зберігання інформації (даних).

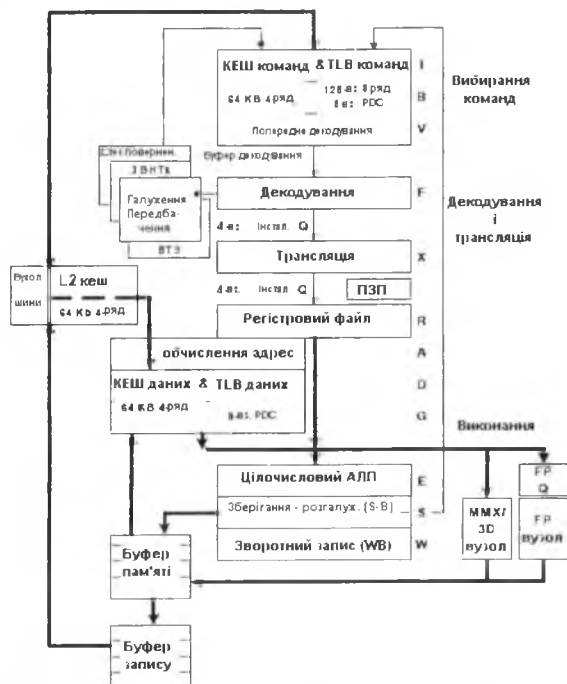


Рис. 2.38. Спрощена блок-схема мікропроцесора Via C3

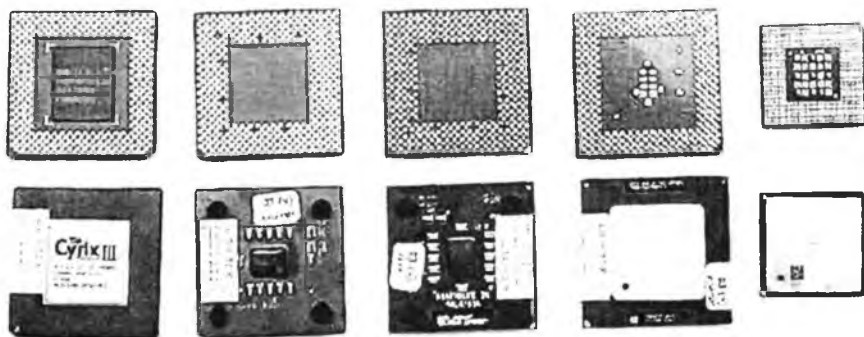


Рис. 2.39. Сучасні мікропроцесори

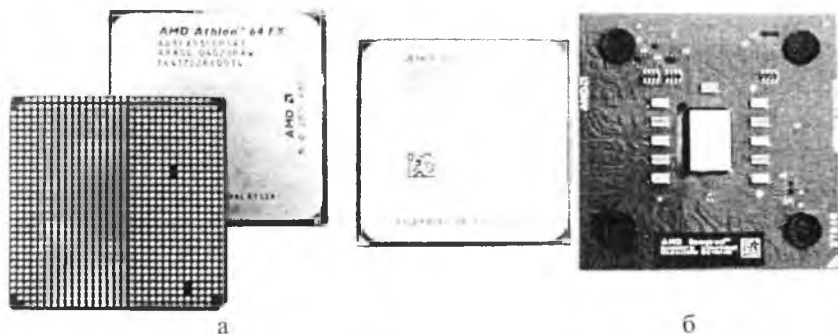


Рис. 2.40. Нові мікропроцесори фірми AMD  
(а – AMD Athlon™ 64 FX, б – AMD Sempron™)

Як і фірма Intel, AMD окрім мікропроцесорів Athlon™ (тактова частота 1,8 – 2,6 ГГц, кеш 512 Кб – 1 Мб) продукує дешевші (у кілька разів!) мікропроцесори Sempron™ (тактова частота 1,5 – 2 ГГц, кеш 256 – 512 Кб), які мають дещо нижчу тактову частоту, аніж найновіші Athlon™ та менший розмір вбудованого кешу L1, L2. Значення тактових частот і розміри вбудованого кешу подані на момент написання книги.

Таблиця 2.7. Мікропроцесори фірми AMD

AMD Athlon 64			AMD Sempron		
Назва	Тактова частота	Гніздо	Назва	Тактова частота	Гніздо
Athlon FX-55	2,6 ГГц	Socket 939	Sempron 3100+	1,8 ГГц	Socket 754
Athlon FX-53	2,4 ГГц	Socket 939	Sempron 3000+	2,0 ГГц	Socket A
Athlon FX-53	2,4 ГГц	Socket 940	Sempron 2800+	2,0 ГГц	Socket A
Athlon FX-51	2,2 ГГц	Socket 940	Sempron 2600+	1,83 ГГц	Socket A
Athlon 64 4000+	2,4 ГГц	Socket 939	Sempron 2500+	1,75 ГГц	Socket A
Athlon 64 3800+	2,4 ГГц	Socket 939	Sempron 2400+	1,67 ГГц	Socket A
Athlon 64 3700+	2,4 ГГц	Socket 754	Sempron 2300+	1,58 ГГц	Socket A
Athlon 64 3500+	2,2 ГГц	Socket 939	Sempron 2200+	1,5 ГГц	Socket A
Athlon 64 3500+	2,2 ГГц	Socket 939	AMD Server/Workstation		
Athlon 64 3400+	2,4 ГГц	Socket 754	Opteron x50	2,4 ГГц	Socket 940
Athlon 64 3400+	2,2 ГГц	Socket 754	Opteron x48	2,2 ГГц	Socket 940
Athlon 64 3200+	2,2 ГГц	Socket 754	Opteron x46	2,0 ГГц	Socket 940
Athlon 64 3200+	2,0 ГГц	Socket 939	Opteron x44	1,8 ГГц	Socket 940
Athlon 64 3200+	2,0 ГГц	Socket 754	Opteron x42	1,6 ГГц	Socket 940
Athlon 64 3000+	2,0 ГГц	Socket 754	Opteron x40	1,4 ГГц	Socket 940
Athlon 64 3000+	1,8 ГГц	Socket 939			
Athlon 64 2800+	1,8 ГГц	Socket 754			



Залежно від кількості контактів та схеми їх розташування розрізняють корпуси мікропроцесорів Socket A, Socket 370, Socket 423, Socket 478, Socket T (775 контактів), Socket 939 (на рис. 2.41 показано встановлення мікропроцесора у гніздо на системній платі). Внаслідок великої кількості (десятки мільйонів) транзисторів у корпусі мікропроцесора та високої частоти перемикання цих транзисторів з одного стану в інший (відкритий/закритий) працюючий мікропроцесор виділяє доволі велику кількість тепла (45-90 Вт для процесорів з тактовою частотою понад 1000 МГц). З метою охолодження мікропроцесора на нього встановлюють радіатор з вентилятором.

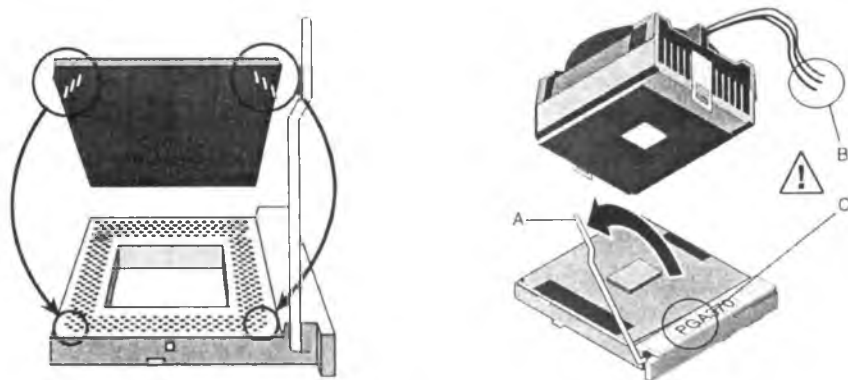


Рис. 2.41. Встановлення мікропроцесора й охолоджувача мікропроцесора;  
А – защіпка гнізда для мікропроцесора, В – дроти живлення вентилятора охолоджувача, С – маркування гнізда для мікропроцесора

Для охолодження мікропроцесорів останнім часом стали використовувати водяні системи охолодження.

Сучасні системи плати обладнані датчиками температури у корпусі системного блоку та на самому мікропроцесорі, покази цих датчиків відстежують допоміжні програми. Якщо температура всередині корпусу системного блоку перевищує 30-35 °С, варто встановити додатковий вентилятор для “виштовхування” теплого повітря з корпусу, температура корпусу мікропроцесора не повинна перевищувати 55 °С. На рис. 2.42 показано правильне встановлення додаткових вентиляторів у корпусі ПЕОМ.

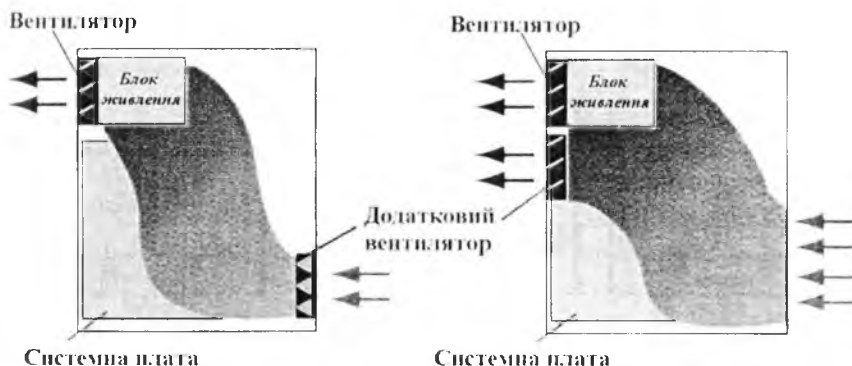


Рис. 2.42. Схеми встановлення додаткових вентиляторів у корпус

Використання багатопроцесорних систем у кластерних обчисленнях, системах зберігання інформації (файл-сервери) створило ґрунт для розробки як багатопроцесорних системних плат (рис. 2.43), так і процесорів з декількома архітектурними одиницями, які проводять обчислення (рис. 2.44).

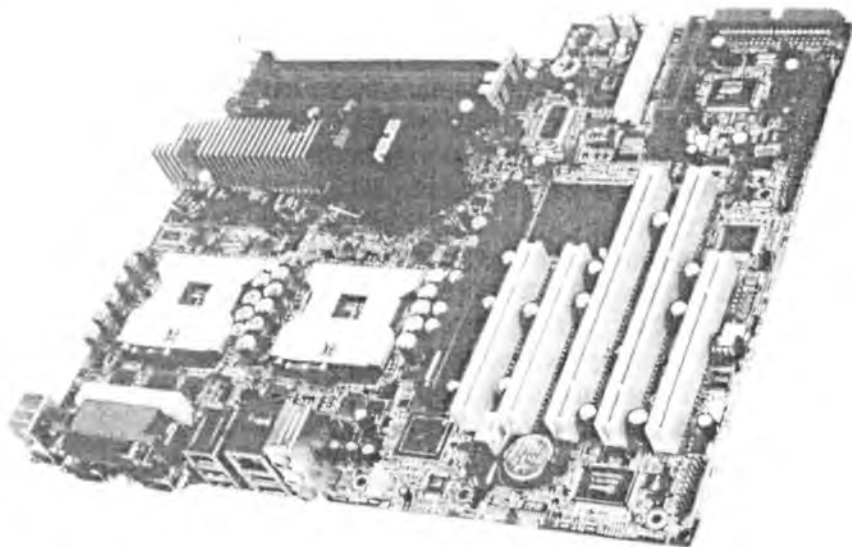


Рис. 2.43. Двопроцесорна системна плата

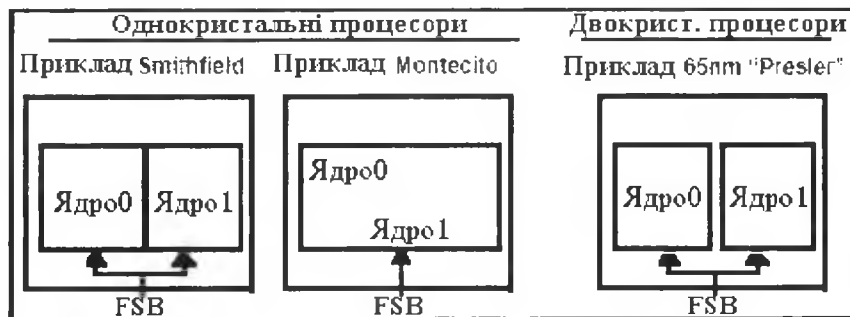


Рис. 2.44. Варіанти побудови двоядерних процесорів

### 2.1.2.11. 64-бітні мікропроцесори

Якщо у 1985 році з появою Intel 80386DX відбувся перехід від 16-бітних мікропроцесорів до 32-бітних, то починаючи з 2001 року відбувається перехід від 32-бітних мікропроцесорів до 64-бітних. Слід зауважити, що 64-бітні RISC-процесори у серверах використовуються з початку 90-х років минулого століття. Однак зараз розпочався процес впровадження 64-бітних мікропроцесорів у виробництво масових IBM-подібних ПЕОМ. Розглянемо зміни, які зумовлені переходом до 64-х біт:

- діапазон цілих чисел збільшився до  $2^{64}-1$  (замість  $2^{32}-1$ );
- розмір адресованої оперативної пам'яті збільшився до  $1,8 \cdot 10^{19}$  байт (замість  $4,3 \cdot 10^9$  байт);
- для забезпечення 64-бітових науково-технічних розрахунків використовується одна 64-бітна команда замість двох 32-бітових (збільшення швидкодії).

Очевидно, що користувач ПЕОМ з 64-бітними мікропроцесорами відчує переваги цих процесорів лише після появи 64-бітних операційних систем і прикладного програмного забезпечення для них (поява ОС Microsoft Vista очікується у 2006 році). Що стосується ОС Linux, то уже кілька збірок (Debian, Red Hat і SuSE) мають версії для 64-бітного Athlon 64. Разом з тим, результати тестування ПЕОМ на 64-бітних мікропроцесорах (у першу чергу Athlon 64) свідчать про збільшення продуктивності ПЕОМ у графічних та ігрових програмах.

### 2.1.2.12. Процесори Power та PowerPC

Архітектура POWER сьогодні – це основа родини робочих станцій і серверів RISC System /6000 компанії IBM. Це традиційна RISC-архітектура: фіксована довжина команд, архітектура “регістр-регістр”, прості способи адресації, великий файл регістрів і триоперандний формат команд. Однак ця архітектура має також кілька додаткових властивостей, що відрізняють її від інших RISC-архітектур.

По-перше, набір команд ґрунтується на ідеї суперскалярного опрацювання. По-друге, архітектура POWER розширена декількома змішаними командами для зменшення часів виконання. Можливо, єдиним недоліком технології RISC порівняно з CISC є те, що іноді вона використовує більшу кількість команд для виконання того самого завдання. З’ясовано, що в багатьох випадках збільшення розміру коду можна уникнути шляхом невеликого розширення набору команд, що зовсім не означає повернення до складних команд, подібних до команд CISC.

Третім фактором, що відрізняє архітектуру POWER від багатьох інших RISC-архітектур, є відсутність механізму “затриманих переходів”. Замість цього архітектура переходів POWER організована для підтримки методики “попереднього перегляду умовних переходів” (branch-lockahead) і методики “згортання переходів” (branch-folding).

Методика реалізації умовних переходів, використовувана в архітектурі POWER, є четвертою унікальною властивістю порівняно з іншими RISC-процесорами.

Платформа PowerPC виникла внаслідок поширення впливу компанії IBM на продукцію компанії Motorola. Вона побудована на платформі RS/6000 у дешевій конфігурації й забезпечує рівень продуктивності, що перевищує можливості сучасних персональних комп’ютерів. В архітектурному плані є кілька варіантів виготовлення процесора PowerPC, що забезпечують потреби портативних виробів і настільних робочих станцій, однак це не перешкоджає застосуванню цих процесорів у великих системах. Першим на ринку був оголошений процесор 601, призначений для використання в настільних робочих станціях компаній IBM і Apple. Потім з’явилися кристали 603 для портативних і настільних систем початкового рівня і 604 для високопродуктивних

настільних систем. Нарешті, процесор 620 створений спеціально для серверних конфігурацій, і розробники сподіваються, що зі своєю 64-бітовою організацією він забезпечить винятково високий рівень продуктивності.

Для розробки архітектури PowerPC з метою задоволення потреб трьох різних компаній (Apple, IBM і Motorola) зі збереженням сумісності з RS/6000, в архітектурі POWER фірми IBM було зроблено декілька змін у таких напрямках:

- спрощено архітектуру з метою її пристосування для реалізації дешевих однокристальних процесорів;
- усунуто команди, що можуть стати перешкодою для підвищення тактової частоти;
- усунуто архітектурні перешкоди суперскалярному опрацюванню та позачерговому виконанню команд;
- додано властивості, необхідні для підтримки симетричного багатопроцесорного опрацювання;
- додано нові властивості, що вважають потрібними для майбутніх прикладних програм;
- забезпечено тривалий час життя архітектури шляхом її розширення до 64-бітової.

Архітектура PowerPC підтримує ту ж базову модель програмування і призначення кодів операцій команд, що й архітектура POWER компанії IBM. У тих місцях, де були зроблені зміни, що могли потенційно перешкоджати процесорам PowerPC виконувати існуючі двійкові коди RS/6000, розставлено "пастки", що забезпечують переривання й емуляцію за допомогою програмного забезпечення. Такі зміни вводили тільки тоді, коли відповідна можливість або використовувалася не дуже часто в кодах прикладних програм, або була ізольована в бібліотечних програмах, які можна просто замінити.

Першим мікропроцесором у родині PowerPC, що повністю підтримує архітектуру PowerPC, є МП PowerPC 603. Він містить п'ять функціональних пристроїв: пристрій переходів, пристрій виконання операцій з цілими числами, пристрій виконання операцій з плаваючою комою, пристрій завантаження/записування і пристрій системних регістрів, а також дві, розташовані на кристалі кеш-пам'яті для команд і даних ємністю 8 Кб кожна. Оскільки PowerPC 603 – суперскалярний мікропроцесор, то він може видавати в ці виконавчі пристрої і завершувати

виконання до трьох команд у кожному такті. Пристрій з плаваючою комою має конвеєрну організацію і виконує операції як з одинарною, так і з подвійною точністю.

Суперскалярний процесор PowerPC 604 забезпечує одночасну видачу до чотирьох команд. У цьому разі паралельно в кожному такті може завершуватися виконання до шести команд. Процесор містить шість виконавчих пристроїв, що можуть працювати паралельно:

- пристрій з плаваючою комою (FPU);
- пристрій виконання переходів (BPU);
- пристрій завантаження/записування (LSU);
- три цілочислових пристрої (IU);
- два одноктактових цілочислових пристрої (SCIU);
- один багатотактовий цілочисловий пристрій (MCIU).

Така рівнобіжна конструкція в сполученні зі специфікацією команд PowerPC, що допускає реалізацію прискореного виконання команд, забезпечує високу ефективність і велику пропускну здатність процесора. У процесорі 604 є окремі пристрої керування пам'яттю й окремі по 16 Кб внутрішні кеші для команд і даних. У ньому реалізовані два буфери перетворення віртуальних адрес у фізичні TLB (окремо для команд і для даних), що містять по 128 рядків. Кеш-пам'яті і буфери TLB використовують для заміщення блоків алгоритм LRU. Процесор 604 має 64-бітову зовнішню шину даних і 32-бітову шину адреси.

Процесор PowerPC 620 призначений для ринку високопродуктивних робочих станцій і серверів. На відміну від своїх попередників, це цілком 64-бітовий процесор. Під час роботи на тактовій частоті 133 МГц його продуктивність оцінюють у 225 одиниць SPECint92 і 300 одиниць SPECfp92, що, відповідно, на 40 і 100% більше від показників процесора PowerPC 604.



### Контрольні запитання

1. Які компоненти входять до системного блока?
2. Які складові входять до системної плати?
3. Яка розрядність системної магистралі у процесорі Intel 8086?
4. Які головні відмінності між процесорами Intel 8086 та Intel 80286?

5. *Яка розрядність шини даних у процесорі Intel 80386SX?*
6. *З якою метою у процесорі запроваджено розширення MMX?*
7. *Назвіть головні відмінності між процесорами Pentium та PentiumPro.*
8. *Яка відмінність між скалярним і суперскалярним процесором?*
9. *Який може бути розмір сторінки у процесорі Pentium?*
10. *Чи підтримує процесор одночасну роботу зі сторінками і сегментами?*
11. *Що означає термін зондовий режим?*
12. *Яка розрядність зовнішньої шини даних?*
13. *З якого процесора розпочинається шосте покоління процесорів фірми Intel?*
14. *У чому полягає суть архітектури процесора Pentium II?*
15. *Як реалізовано трирівневий кеш у процесорі Pentium III?*
16. *Для чого застосовано внутрішнє помноження частоти у процесорі Pentium III?*
17. *У чому полягає суть мікроархітектури NetBurst?*
18. *Яка відмінність між скалярним і конвексним способом виконання команд?*
19. *Яка розрядність зовнішньої шини даних у процесорі i860?*
20. *Як організована робота пам'яті у МП i860?*
21. *Що відрізняє архітектуру POWER від багатьох інших RISC-архітектур?*

### **2.1.3. Оперативна пам'ять**

В оперативній пам'яті ПЕОМ розміщують програми, які виконує мікропроцесор, та інформацію (дані), яку опрацьовує мікропроцесор. Від розміру і швидкодії оперативної пам'яті суттєво залежить швидкодія вашої системи. Оперативна пам'ять сучасних ПЕОМ будується на основі мікросхем динамічної пам'яті (слід зауважити, що оперативна пам'ять на конденсаторах була ще у проекті Атанасова-Беррі).

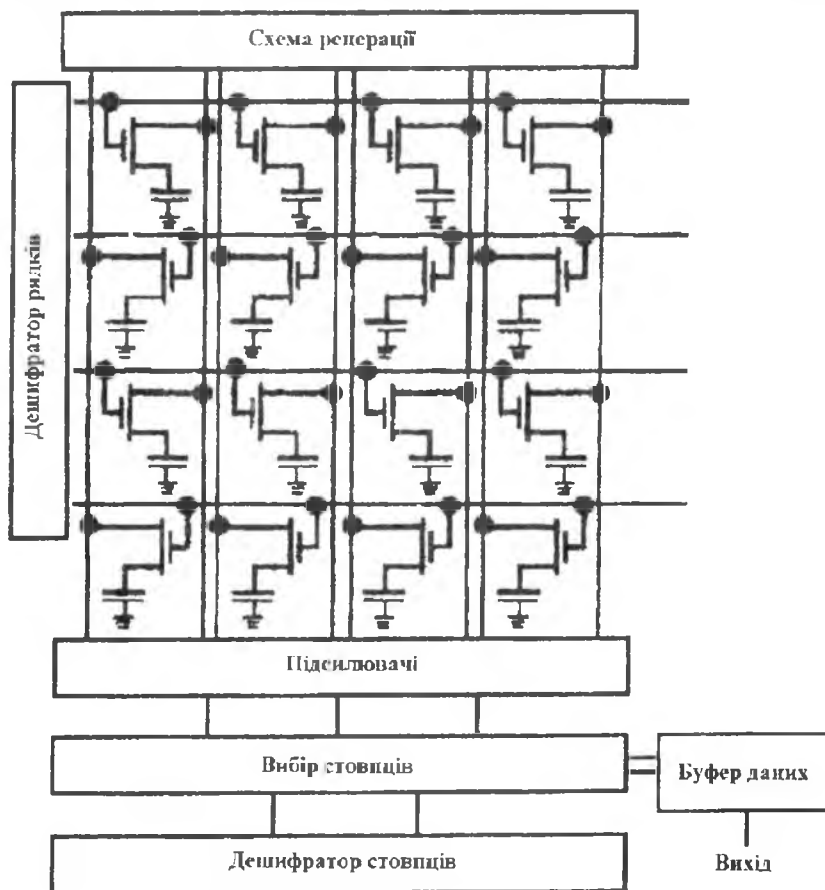


Рис. 2.45. Організація динамічної пам'яті

У кожному конденсаторі зберігається один біт інформації (конденсатор заряджений – двійкова “1”, розряджений – “0”). Унаслідок неідеальності ізолятора між обкладками конденсатора всі заряджені конденсатори врешті-решт розрядяться. Для того, щоб уникнути втрати інформації внаслідок розряду конденсаторів, заряджені конденсатори періодично (не пізніше ніж за 50 мілісекунд) підзаряджають (регенерують). Це завдання покладають на систему регенерації динамічної пам'яті – система побітно вичитує інформацію, записану в динамічну пам'ять і, якщо у відповідній комірці була записана двійкова “1”,



дозаряджає конденсатор до номінального рівня. У перших ІВМ-подібних ПЕОМ мікросхеми динамічної пам'яті встановлювалися прямо на системну плату. Згодом стали використовувати невеличкі платки (SIMM і DIMM-модулі), на які впаювали кілька мікросхем динамічної пам'яті. На рис. 2.46 подані зображення модулів пам'яті, а на рис. 2.47 – встановлення DIMM-модуля.

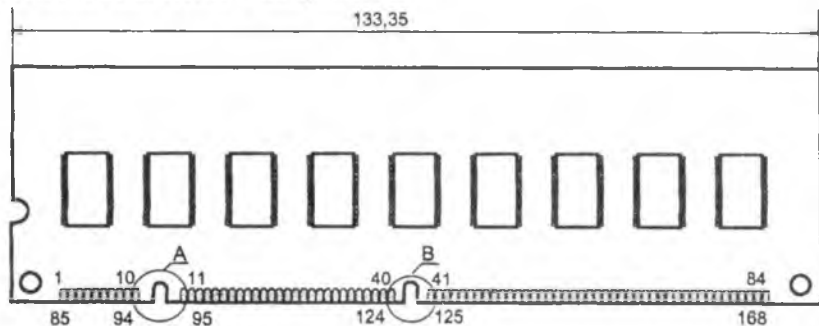


Рис. 2.46. 168-контактні модулі DIMM (Dual in line Memory Module)

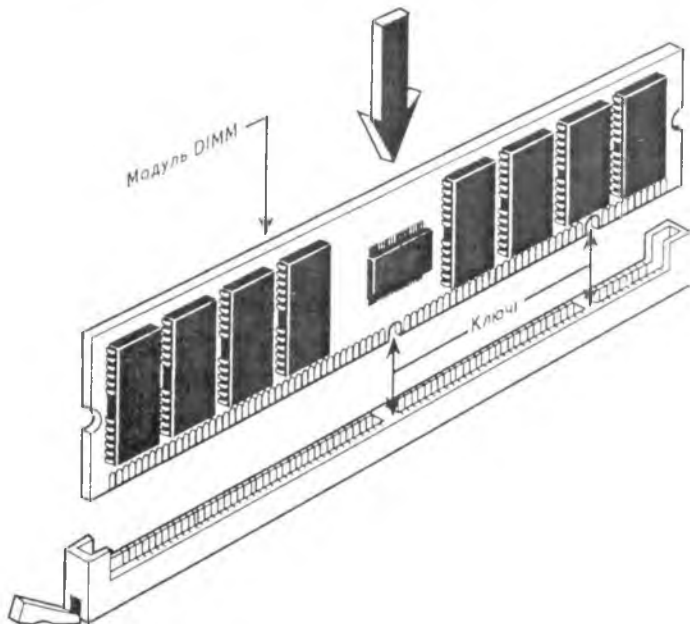


Рис. 2.47. Встановлення DIMM-модуля

Кількість гнізд для встановлення модулів динамічної пам'яті залежить від конструкції системної плати. За методами доступу до зберезуваної інформації розрізняють такі типи динамічної пам'яті:

- FPM (Fast Page Mode) – застаріло;
- EDO (Extended Data Out) – застаріло;
- SDRAM (PC66, PC100, PC133) – синхронна динамічна пам'ять;
- DDR SDRAM (DDR 160, DDR 200, DDR 266, DDR 333) – синхронна динамічна пам'ять з обміном по передньому і задньому фронту тактового імпульса;
- DDR2 – синхронна динамічна пам'ять з чотирма обмінами на тактовому імпульсі;
- Rambus DRAM (RDRAM 400, RDRAM 533, RDRAM 800, RDRAM 1066) – динамічна пам'ять з окремими шинами обміну інформацією і керування.

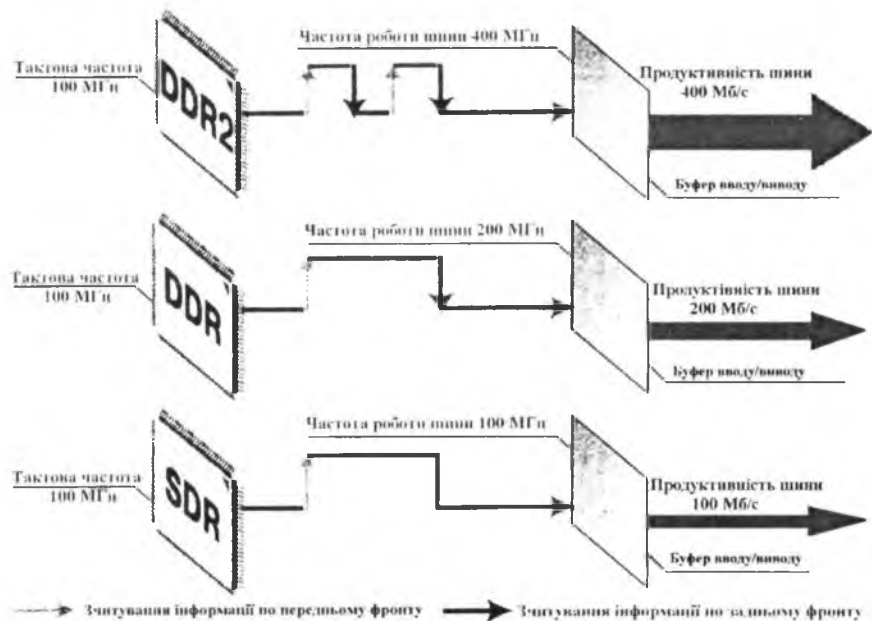


Рис. 2.48. Принципи роботи мікросхем динамічної пам'яті

Тип динамічної пам'яті, яку можна встановити на системну плату, залежить від мікросхем системної логіки. Цю обставину слід враховувати при модернізації ПЕОМ.

Динамічна пам'ять не є єдиним типом оперативної пам'яті. У різних поколіннях ЕОМ використовувалися різні принципи побудови оперативної пам'яті – ртутні трубки, осцилографічні трубки, тригери, феритові кільця і т. ін. Поява більш містких і дешевших пристроїв оперативної пам'яті приводила до витіснення з ринку менш ефективних видів пам'яті. Наприкінці 2004 р. розпочався промисловий випуск мікросхем фероелектричної пам'яті (фірма Ramtron). Слід наголосити на тому, що інформація у мікросхемах фероелектричної пам'яті зберігатиметься після вимкнення живлення (як тут не згадати про пам'ять на феритових кільцях).

Останніми роками активно ведуться дослідницькі роботи по створенню запам'ятовуючих пристроїв на вуглецевих нанотрубках, які були створені у 1991 р.



Рис. 2.49. Циліндричний фуллерен – нанотрубка

Фірма Nantero розробила прототип запам'ятовуючого пристрою на нанотрубках.



Рис. 2.50. Запам'ятовуючий пристрій на нанотрубках

Якщо до електрода і контакта однієї з нанотрубок під'єднати різноміснні полюси джерела напруги, то за рахунок сили взаємодії електричних зарядів нанотрубка прогнеться і торкнеться до електрода.

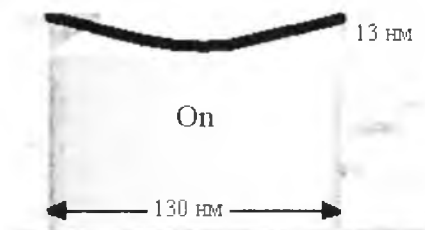


Рис. 2.51. Запис двійкової одиниці у нанокмірку

Після зняття напруги нанотрубка залишається деформованою завдяки силі ближньої взаємодії Ван-дер-Ваальса, яка пропорційна  $r^{-6}$  і проявляється лише на малих відстанях між об'єктами (відстань від нанотрубки у недеформованому стані до електрода складає лише 13 нанометрів). На рис. 2.52 подана залежність потенційної енергії системи “нанотрубка – електрод” від відстані між ними. Мінімум потенційної енергії системи відповідає стійкому стану системи. Слід наголосити на тому, що пам'ять на нанотрубках є енергонезалежною – записана у неї інформація зберігається після вимкнення живлення.



Рис.2.52. Стійкий стан системи “нанотрубка – електрод”

Серійний випуск мікросхем оперативної пам'яті на нанотрубках очікується через кілька років.

Від частоти зовнішньої шини процесора (Front Side Bus), характеристик північного моста системної логіки і модулів оперативної пам'яті залежить швидкість читання інформації з оперативної пам'яті. На рис. 2.53 подана стовпчикова діаграма швидкості читання інформації з оперативної пам'яті для різних процесорів, наборів системної логіки і типів оперативної пам'яті (діаграма побудована програмою aida32.exe).

CPU		Motherboard	Chipset	Memory
3180 MB/s	P4-2.1A	Abit TH7II	i850	PC1066 Dual RDRAM
2600 MB/s	P4-2.1A	Abit TH7II	i850	PC800 Dual RDRAM
2540 MB/s	P4-2.4A	Asus P4T533-C	i850E	PC800 Dual RDRAM
2480 MB/s	AthlonXP 1.7 GHz	Abit KX7-333	KT333	PC2900 DDR SDRAM
2480 MB/s	P4-2.1A	Abit SR7-8X	SiS64E	PC2700 DDR SDRAM
2460 MB/s	Xeon-2.4A	Intel SE7500CW2	IE7500	PC1600 Dual DDR
2430 MB/s	Athlon-1.4	MSI KT3 Ultra	KT333	PC2700 DDR SDRAM
2050 MB/s	AthlonXP-1800+	Epox 8K3A+	KT333	PC2700 DDR SDRAM
2030 MB/s	P4-2.0A	Gigabyte GA-8IE533	i845E	PC2100 DDR SDRAM
2020 MB/s	AthlonXP-1700+	Asus A7V8X	KT400	PC2700 DDR SDRAM
2020 MB/s	P4-2.0A	Asus P4B266	i845D	PC2100 DDR SDRAM
2010 MB/s	AthlonXP-2000+	Asus A7V333	KT333	PC2700 DDR SDRAM
1990 MB/s	AthlonXP-1800+	Asus A7N8X	nForce615D	PC2100 Dual DDR
1930 MB/s	Celeron4-1.7	DFI NS35-SL	SiS650GX	PC2100 DDR SDRAM
1910 MB/s	Celeron4-1.8	MSI 645 Ultra	SiS645	PC2100 DDR SDRAM
1910 MB/s	P4-2.2A	MSI 645E Max2	SiS645DX	PC2100 DDR SDRAM
1880 MB/s	AthlonXP-1800+	Epox 8KHA+	KT266A	PC2100 DDR SDRAM
1870 MB/s	AthlonXP-1800+	ECS K7S6A	SiS745	PC2100 DDR SDRAM
1843 MB/s	This Computer	ECS K7VTA3 v3	KT333	
1840 MB/s	AthlonXP-1800+	MSI KT3 Ultra	KT333	PC2100 DDR SDRAM
1830 MB/s	AthlonXP-1800+	Asus A7V333	KT333	PC2100 DDR SDRAM
1820 MB/s	AthlonXP-1700+	Azza KT33-BV	KT333	PC2100 DDR SDRAM
1550 MB/s	AthlonXP-1600+	Abit KG7	AMD-760	PC2100 DDR SDRAM
1410 MB/s	Duron-900	Chaintech 7VJDA	KT266A	PC1600 DDR SDRAM
1080 MB/s	Duron-600	Tysn Tiger MP	AMD-760MP	PC1600 DDR SDRAM
1030 MB/s	PIII-1133S	Asus TUSL2-C	i815EP	PC133 SDRAM
1030 MB/s	PIII-733EB	MSI 815EM Pro	i815E AGP	PC133 SDRAM
1030 MB/s	Celeron-1333S	ECS P6S5AT	SiS635	PC133 SDRAM
1000 MB/s	PIII-1200S	Asus TUV4X	ApolloPro133T	PC133 SDRAM
990 MB/s	P4-1.6	Asus P4B-MX	i845	PC133 SDRAM
910 MB/s	PIII-733EB	MSI 815EM Pro	i815E Int.	PC133 SDRAM
900 MB/s	Duron-800	Asus A7V-E	KT133A	PC133 SDRAM

Рис. 2.53. Стовпчикова діаграма швидкості читання інформації з оперативної пам'яті

### 2.1.4. Кеш-пам'ять

З метою збільшення швидкодії при звертанні процесора до оперативної пам'яті в архітектурі 32-розрядних процесорів реалізована така ієрархія пам'яті, яка передбачає наявність відносно великої ємності і малої швидкодії динамічної DRAM (Dynamic RAM) та меншої ємності і більшої швидкодії кеш-пам'яті, або статичної SRAM (Static RAM).

Термін “кеш” (cache) у нашому випадку відповідає значенню “схованка” і секрет цієї схованки полягає в тому, що спеціальний контролер кешу може передбачати використання процесором певної частини оперативної пам’яті і наперед завантажувати її у швидкодіючу кеш-пам’ять. У сучасних комп’ютерах кеш-пам’ять будується за дво- або трирівневою схемою. Кеш першого рівня (L1 Cache) вбудований безпосередньо в мікросхему ядра мікропроцесора, починаючи з i486. Кеш другого рівня (L2 Cache) встановлювався на системній платі і приєднувався до спеціальної внутрішньої шини процесора. Починаючи з процесора Р6 кеш другого рівня монтують у корпусі мікропроцесора. Ємність кешу вищих рівнів сягає сотень Мб, а швидкодія приблизно удвічі менша від кешу першого рівня. Кеш третього рівня встановлюють на системній платі, або в корпусі мікропроцесора (використовують при побудові серверних і кластерних систем).

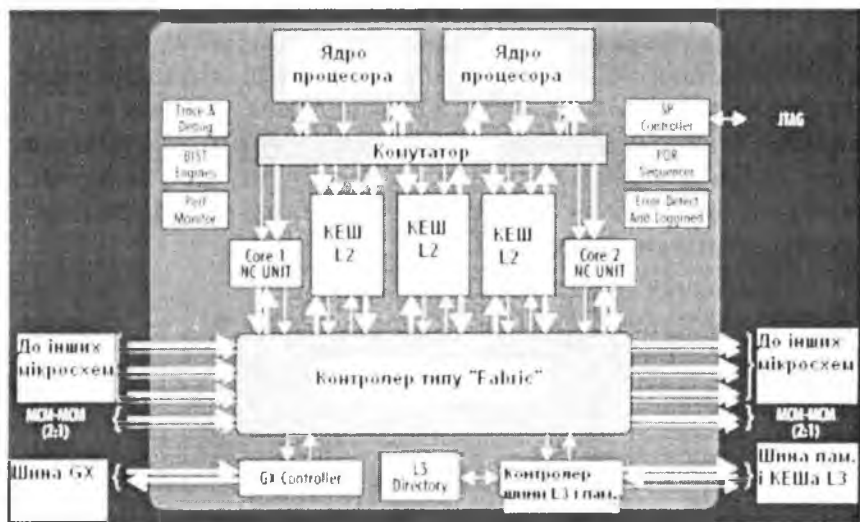


Рис. 2.54. Структурна схема взаємодії кеш-пам'яті з обладнанням мікропроцесора на прикладі IBM Power4

## Контрольні запитання

1. Поясніть призначення оперативної пам'яті.
2. Що таке динамічна пам'ять?

3. Чи зберігається інформація у мікросхемах динамічної пам'яті після вимкнення живлення ПЕОМ?
4. Які фактори визначають швидкість обміну інформацією мікропроцесора із оперативною пам'яттю?
5. Поясніть відмінності між модулями пам'яті DDR і DDR2.
6. Як модулі оперативної пам'яті встановлюють на системну плату?
7. Поясніть призначення кеш-пам'яті.

### 2.1.5. Система шин

Для взаємодії мікропроцесора зі складовими системної плати використовують шини даних (ШД), адреси (ША) та керування (ШК). По шині даних передається інформація між складовими системного блока. Інформація на шині адреси визначає, кому призначена інформація. Шину керування використовують для синхронізації процесів обміну інформацією та керування складовими системного блока. Для приєднання варіативної частини обладнання (звукова карта, графічний контролер, контролер локальної мережі і т. ін.) використовують роз'язки плат розширення, які виконані у відповідності до певного стандарту. Усі стандарти системних шин здійснюють паралельну передачу інформації, лише стандарт PCI Express – послідовну. У таблиці подані характеристики стандартів системних шин.

Стандарт	Розрядність шини даних (біт)	Тактова частота (МГц)	Примітка
ISA-8	8	8	Застаріло
ISA-16	16	8,16	
EISA	32	8.33	Дорога, обмежене використання
MCA	16/32	10	PS/2
VLB	32/64	33-50	Застаріло
PCI	32/64	33,66	
PCI Express	32/64	66	Послідовна шина, базується на парі послідовних каналів передачі інформації (один на вивід, другий на ввід) з швидкістю передачі 200Мб/с. Кількість таких пар визначає варіант шини x1 – одна пара, x2 – дві пари і т. д. [x1, x2, x4, x8, x12, x16, x32]. Більш докладну інформацію можна знайти на сайті <a href="http://www.express-lane.org">www.express-lane.org</a>

Інтерфейс PCI Express покликаний замінити інтерфейси PCI та AGP (наразі можна придбати системні плати або з портом AGP, або з інтерфейсом PCI Express). У 2004 р. розпочалося виробництво системних плат з цим інтерфейсом. Висока пропускна здатність забезпечується (максимально  $32 \cdot 256$  Мб/с) розбиттям потоку даних на певну кількість потоків із послідовною передачею інформації (від 1 до 32). Швидкість передачі інформації в одному потоці в одному напрямку складає 256 Мб/с. На рис. 2.55 подана структура одного каналу PCI Express, а на рис. 2.56 – схема формування потоку бітів у каналі.

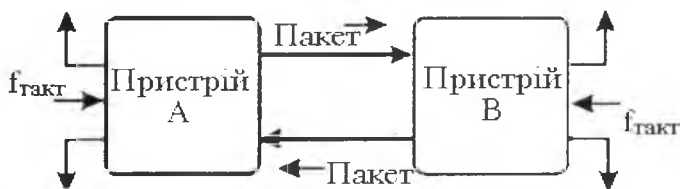


Рис. 2.55. Структура одного каналу (x1) інтерфейсу PCI Express

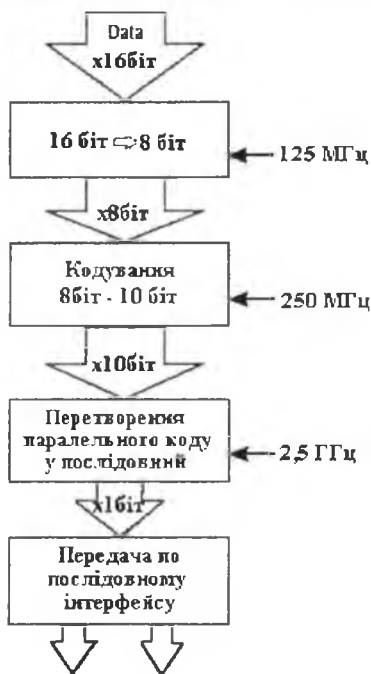


Рис. 2.56. Схема формування потоку бітів у каналі інтерфейсу PCI Express



Специфікація AGP (Accelerated Graphics Port) була запропонована фірмою Intel у 1996 р. Порт AGP призначений для з'єднання процесора з графічним контролером. Графічні процесори, які використовують контролери за специфікацією AGP, можуть споживати потужність до 110 Вт. У таблиці подані технічні характеристики специфікацій AGP.

Специфікація	Базова частота, МГц	Кількість передач на один такт базової частоти	Швидкість передачі інформації, Мбайт/с	Напруга живлення, В
AGP*1	66	1	266	3.3
AGP*2	66	2	533	1.5
AGP*4	66	4	1066	1.5
AGP*8	66	8	2132	1.5

На рис. 2.57 зображені розняття плат розширення різних стандартів та їх розташування на системній платі відносно задньої стінки ПЕОМ.

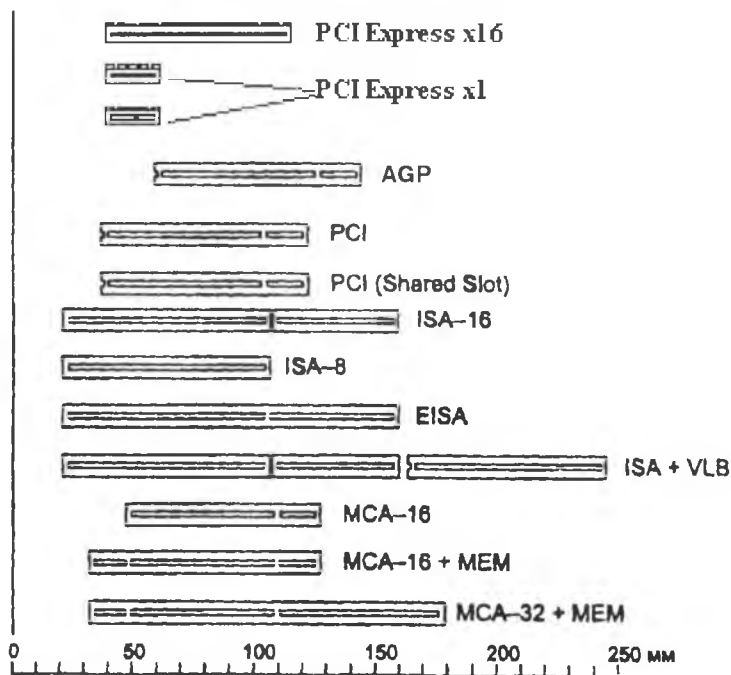


Рис. 2.57. Розташування розняття системної шини відносно задньої стінки системного блоку



## Контрольні запитання

1. Поясніть призначення системної шини.
2. Перерахуйте стандарти системних шин.
3. Для чого призначена шина AGP?
4. Для чого призначена шина PCI Express?
5. Які стандарти системних шин використовують у сучасних системних платах?
6. Як здійснюється взаємодія системної шини PCI, AGP, PCI Express із мікропроцесором?

### 2.1.6 Базова система вводу-виводу (BIOS)

**BIOS** (Basic System Input-Output) – базова система вводу-виводу призначена для зняття залежності операційної системи від апаратних особливостей конкретної системної плати. **BIOS** розміщують у перепрограмованому постійному запам'ятовуючому пристрої. **BIOS** містить у собі програмні засоби, які підтримують стандартні ресурси ПЕОМ, самотестування обладнання (**POST**), задання конкретної конфігурації ПЕОМ з клавіатури (**Setup**) та виклик програми для завантаження операційної системи (ОС) з системного диска.

### 2.1.7 Програма POST самотестування ПЕОМ

Після увімкнення живлення системного блоку схема контролю напруги +5В виробляє сигнал **PG** (Power Good), якщо її значення відповідає номінальному. Сигнал **PG** ініціює виконання програми **POST** (Power On Self Test) – програми самотестування ПЕОМ після ввімкнення живлення. Як правило, тест **POST** складається з кроків:

- тестування регістрів мікропроцесора;
- перевірка контрольної суми **BIOS**;
- перевірка й ініціалізація таймера (після цього кроку можливі звукові повідомлення про результати тестування);
- перевірка й ініціалізація контролера ПДП;
- перевірка регенерації пам'яті;
- тестування 64К нижньої пам'яті;

- завантаження векторів переривань і стеку в нижню область пам'яті;
- ініціалізація відеоконтролера (після цього кроку діагностичні повідомлення виводяться на екран);
  - тестування пам'яті;
  - тестування клавіатури;
  - тестування **CMOS** і годинника;
  - ініціалізація **COM** і **LPT**-портів;
  - ініціалізація і тест контролера гнучких магнітних дисків;
  - ініціалізація і тест контролера жорстких магнітних дисків;
  - сканування області розширення **BIOS**;
  - завантаження операційної системи.

У випадку виникнення несправностей аналіз повідомлень тесту дозволяє визначити несправну складову ПЕОМ, однак кодування повідомлень (звукове і текстове) залежить від версії **BIOS**. Розглянемо звукові сигнали, які для більшості версій **BIOS** мають однакове значення:

- неперервний звуковий сигнал – несправність системної плати;
- періодично повторюваний довгий гудок – несправна оперативна пам'ять;
- періодично повторювані один довгий, три короткі гудки – несправний графічний контролер;
- періодично повторюваний короткий гудок при працюючій ПЕОМ – перегрів мікропроцесора.

Незважаючи на особливості конкретних версій **BIOS**, цифрові коди помилок і типи несправностей можна задати наступною таблицею.

Діапазон кодів помилок	Несправність
100 – 199	несправності системної плати
200 – 299	несправності оперативної пам'яті
300 – 399	несправності клавіатури
400 – 499	несправності монохромного дисплею
500 – 599	несправності кольорового дисплею
600 – 699	несправності пристрою для роботи з ГМД
901	несправний <b>Lpt1</b>
1001	несправний <b>Lpt2</b>
1401, 1404	несправний пристрій друку
1700 – 1799	несправності ЖМД і контролера ЖМД

### 2.1.8. Програма Setup задання параметрів обладнання

Програма встановлення параметрів обладнання **Setup** є складовою частиною базової системи вводу/виводу (**BIOS**). Параметри обладнання, задані за допомогою цієї програми, зберігаються у пам'яті з малим споживанням струму (**CMOS**-пам'ять). Живлення цієї пам'яті здійснюється від батарейки, яку встановлюють на системну плату. Для запуску програми **Setup** потрібно під час виконання тесту **POST** натиснути певні клавіші (**Del** або **Ctrl+Alt+Esc**, **F1** або **F2**). Конкретна комбінація клавіш для входу у **Setup** залежить від версії **BIOS**. Для деяких системних плат потрібна спеціальна системна дискета (наприклад **IBM-PS/2**), для інших потрібно встановити місток (jumper) на системній платі у певне положення.

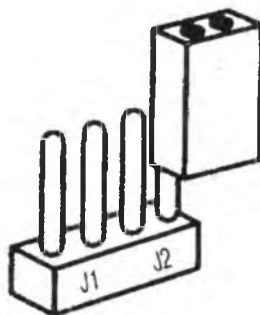


Рис. 2.58. Місток для “закорочування” контактів

Меню програми також залежить від версії базової системи вводу/виводу. Як приклад розглянемо **Phoenix – Award BIOS Setup**.

Меню програми складається із п'яти підменю (інші версії можуть мати інші найменування підменю, однак загалом більшість програм **Setup** є подібними). Вибір окремих пунктів меню програми здійснюють за допомогою клавіш керування курсором. Розглянемо окремі пункти меню:

**Main** – використовують для задання дати, часу, параметрів ЖМД, ГМД. Останні версії програми **Setup**, як правило, використовують опцію **Auto**;

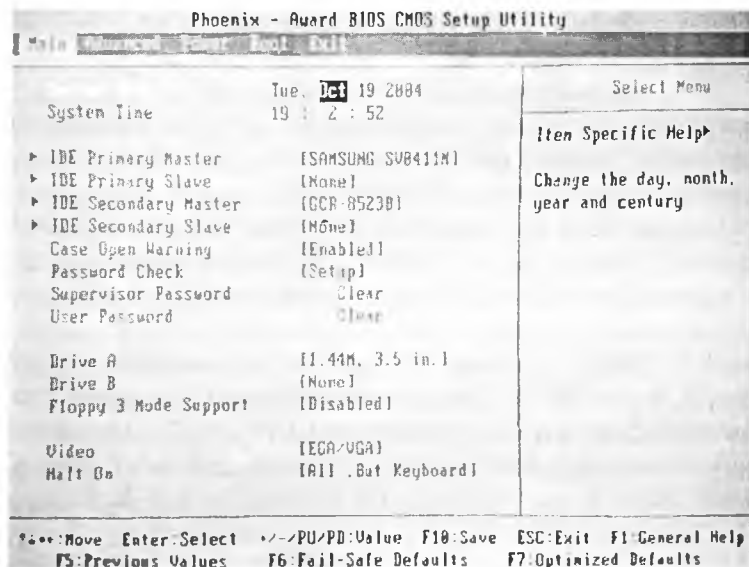


Рис. 2.59. Вікно програми Setup із вибраним підменю “Main”

**Advanced** – у цьому пункті меню задають додаткові (розширені) параметри BIOS. Більшість параметрів цього пункту меню доцільно не змінювати після задання їх на комп’ютерній фірмі, однак окремі параметри цього пункту часом все ж таки доведеться змінювати самому:

**Anti Virus Protection** – Enable/Disable (відстеження спроб внесення змін у завантажувальний сектор – boot sector ЖМД, при спробі комп’ютерного вірусу змінити цей запис на дисплей виводиться повідомлення про спробу зміни інформації у цьому секторі. При інсталяції програмного забезпечення рекомендується встановити цей пункт у положення Disable);

**Peripherals** – у цьому підпункті меню задають параметри контролерів периферійних пристроїв, вбудованих у системну плату. Змінювати інформацію, задану тут, радимо лише у разі потреби. Окремо звернемо вашу увагу на рядок Onboard Paralel Mode (рис. 2.60), який використовують для задання режиму роботи інтерфейсу Centronics. Вибір режиму істотно залежить від під’єданого до цього інтерфейсу обладнання. В інструкції до придбаного пристрою вказують, який режим роботи інтерфейсу Centronics потрібен для цього пристрою.

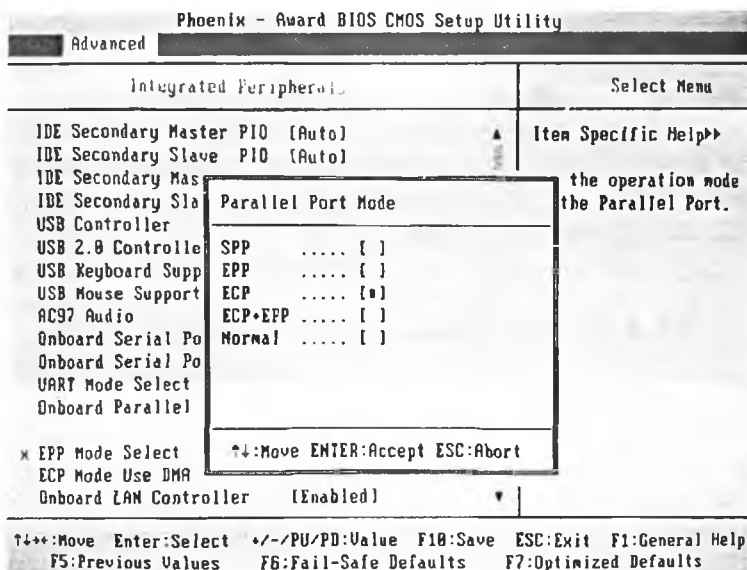


Рис. 2.60. Вікно програми Setup із вибраним підменю “Advanced” та підпунктом “Parallel Port Mode”



Рис. 2.61. Вікно програми Setup із вибраним підменю “Boot”

**Power** – використовують для задання параметрів режиму енергозбереження;

**Boot** – використовують для задання черговості пошуку системних файлів на дискових пристроях, пристроях архівування інформації і в локальній мережі. На рис. 2.61 подано вікно програми **Setup** із вибраним підменю “**Boot**”.

**Exit** – вихід із програми **Setup**.

### 2.1.9 Методи запису і модернізації BIOS

Для запису **BIOS** використовують такі види ПЗПр:

- **ROM** (Read Only Memory, пам’ять лише читання), **PROM** (Programmable ROM, програмована пам’ять лише читання) – постійний запам’ятовуючий пристрій одноразового запису (ПЗПр.);

- **EPROM** (Erasable PROM, пам’ять лише читання з оптичним витиранням інформації) – багаторазово перепрограмований ПЗПр.;

- **EEPROM** (Electrical EPROM, ПЗПр. з електричним стиранням інформації) – багаторазово перепрограмований ПЗПр.;

- **Flash Memory** (флеш-пам’ять).

**EPROM**, **EEPROM** та **Flash Memory** дозволяють модернізувати базову систему вводу/виводу шляхом запису нової версії **BIOS**.

Для перезапису вмісту **EPROM** застосовують програматор (для запису інформації) з ультрафіолетовою лампою (для витирання інформації). Такі ПЗПр. використовувались у старіших моделях ПЕОМ. Згодом виробники системних плат перейшли до використання ПЗПр. з електричним витиранням інформації (**EEPROM**). Унаслідок цього перезапис **BIOS** став можливий безпосередньо на ПЕОМ (In System Write). Цю можливість використовують деякі вірусні програми, наприклад **chih**.

Версія **BIOS** вибирається виробником системної плати, серед найбільш поширених – базові системи вводу/виводу фірм **Award**, **AMI** та **Phoenix**. На Web-сайтах цих компаній можна знайти найновіші версії **BIOS**, які можна встановити на ПЕОМ. Зауважимо, що модернізація **BIOS** може виявитись тривалим і непростим процесом, для його виконання потрібен кваліфікований персонал, а спроба виконати модернізацію самостійно може скінчитися катастрофою!

Плати контролерів можуть мати ПЗПр. з програмами підтримки (розширення BIOS), які виявляються і враховуються операційною системою при виконанні тесту POST. Розширення BIOS використовують контролери жорстких дисків, мережеві контролери з віддаленим завантаженням ОС, графічні контролери.

Фахівці Intel ведуть розробку Extensible Firmware Interface (EFI), яка покликана замінити BIOS. EFI написана мовою Сі і зберігається на жорсткому магнітному диску в секторах, які недоступні програмному забезпеченню.

### ? Контрольні запитання

1. Поясніть значення терміну BIOS.
2. З яких складових частин складається BIOS?
3. Як розпочинається робота ПЕОМ після увімкнення живлення?
4. Для чого призначена програма POST?
5. Для чого призначена програма Setup?
6. Які методи використовують для запису і модернізації BIOS у сучасних системних платах?
7. Чи існує можливість ураження BIOS вірусами?

## 2.1.10. Системні засоби вводу-виводу

### 2.1.10.1. Клавіатура

Клавіатуру використовують для вводу інформації в ПЕОМ. Клавіатуру приєднують до контролера на системній платі за допомогою октального роз'ємця (застаріло), роз'ємця PS/2, роз'ємця інтерфейсу USB.

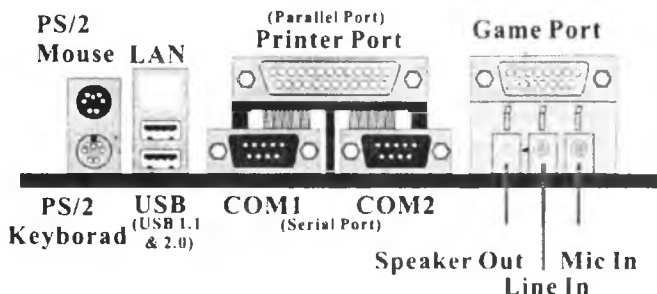


Рис. 2.62. Задня стінка системного блоку (ATX) з роз'ємцями



Клавіатура складається з клавішного поля (рис. 2.63), контролера і кабеля приєднання до системного блоку (шнура). При натисканні клавіші замикаються відповідні контакти (або змінюються інші електричні параметри клавішного елемента – опір, ємність і т. ін.), контролер клавіатури відслідковує (сканує) такі події і передає інформацію про це у системний блок.



Рис. 2.63. Клавішне поле клавіатури

За конструкцією клавішного елемента розрізняють клавіатури плівкові (найдешевші), з механічними перемикачами, ємнісні, магнітодiodні, герконові і т.ін. На рис. 2.64 подана будова клавішного елемента генераторної (ємнісної) клавіатури і клавіатури з замиканням електричного контакту, а на рис. 2.65 – зображення плівок плівкової клавіатури. У цьому прикладі використано три плівки (нижня і верхня з провідниками і контактними майданчиками, середня з отворами в місцях розташування контактних майданчиків). У недеформованому стані контактні майданчики верхньої і нижньої плівок не торкаються один одного. При натисканні клавіші верхня плівка прогинається і замикаються відповідні контактні лінії. Замикання контактних ліній відслідковує контролер клавіатури.

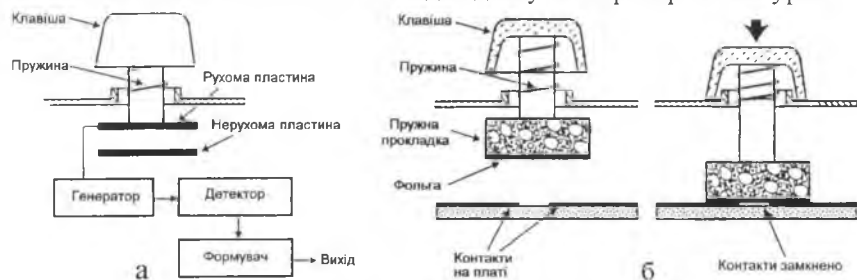


Рис. 2.64. Клавішні елементи:

а – ємнісної клавіатури; б – клавіатури із замиканням контактів

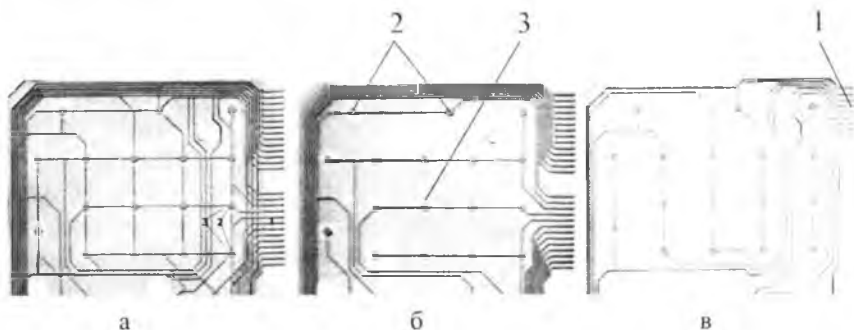


Рис. 2.65. Плівки плівкової клавіатури: а – три плівки, накладені одна на одну; б – нижня і середня плівки; в – верхня плівка: 1 – контакти для присіднання контролера клавіатури; 2 – контактні майданчики; 3 – отвори у середній плівці

Довготривале використання ігор з керуванням від клавіатури, як правило, призводить до відмови окремих клавішних елементів.

Для зменшення навантаження на руки при довготривалій роботі на клавіатурі використовують підставки для рук та ергономічні клавіатури.

Деякі виробники пропонують бездротові клавіатури (з передаванням інформації інфрачервоними променями або радіохвилями). Таку клавіатуру можна досить вільно переставляти на робочому столі без ризику зачепити щось шнуром клавіатури. На рис. 2.66 подані зображення клавіатури з додатковими клавішами і бездротова клавіатура й миша.



Рис. 2.66. Клавіатура з додатковими клавішами і бездротова клавіатура

### 2.1.10.2. Координатно-вказівні пристрої (mouse, track-ball, touch-pad)

У всіх сучасних моделях ПЕОМ координатно-вказівний пристрій миша використовують як стандартний інтерфейс взаємодії з ПЕОМ. Конструктивно миша складається з кулі, яка обертається при переміщенні по поверхні килимка. Обертання кулі передається на два перпендикулярно розміщених валки. На осі валків насаджено прорізні диски. Світловий потік від світлодіода переривається при обертанні диска (переміщенні миші). Фотоприймач, розміщений з іншого боку диска, перетворює зміни світлового потоку в послідовність електричних імпульсів.



Рис. 2.67. Конструкція механічної миші



Рис. 2.68. Координатно-вказівні пристрої миша

Імпульси від фотоприймачів та сигнали від клавіш миші опрацьовуються мікроконтролером миші, опрацьована інформація передається у системний блок. Виробники випускають миші з кількістю клавіш від двох до п'яти. Для зменшення кількості переміщень мишки по килимку в конструкцію деяких мишок введено коліщатко прокручування (одне або два). Коліщатко прокручування дозволяє переміщувати документ у вікні програми без вказування на піктограми прокручування. На рис.2.68 подані фотографії мишок з різною кількістю елементів керування.

Маніпулятори миша приєднуються до ПЕОМ через послідовні порти COM1, COM2,

PS/2, інтерфейс USB, оптичний канал та радіоканал (бездротові мишки).

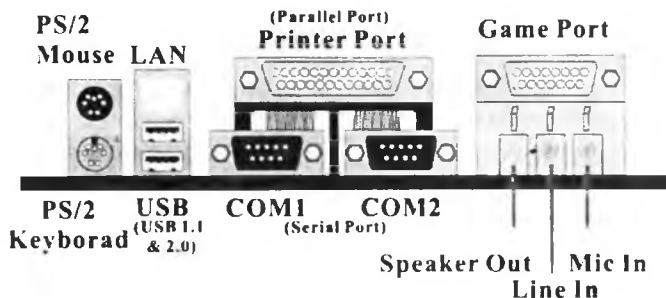


Рис. 2.69. Задня стінка системного блоку з роз'ємними для під'єднання миші

Переміщення маніпулятора миші вимірюють у кроках. Крок – це мінімальне переміщення миші, що реєструється її давачами. Як правило, роздільна здатність давачів переміщення миші становить близько 8 кроків/мм, хоч існують більш чутливі моделі, що забезпечують точність до 16 кроків/мм. Опрацювання інформації про переміщення миші та натискання її клавіш здійснюється спеціальною програмою – драйвером, користувацькі характеристики миші істотно залежать від цієї програми.

Окрім механічних мишок (з кулькою і валками) виробники пропонують оптичні миші – промінь від червоного світлодіода відбивається від поверхні стола і потрапляє на мініатюрну відеокамеру. Зображення ділянки стола, на яку скеровано відеокамеру, з великою швидкістю (від 1500 до 6000 знімків у секунду) опрацьовується цифровим сигнальним процесором. Зміни зображення цифровий сигнальний процесор перераховує у переміщення мишки. Чутливість оптичних мишок складає не менше 16 кроків/мм.

Маніпулятор track-ball можна розглядати як перекинуту догори дном мишку, в якій кульку обертають за допомогою пальця. Track-ball, як правило, використовують у переносних ПЕОМ, однак відомі спроби поєднання маніпуляторів миша і track-ball в одному корпусі (рис.2.70) та вбудовування track-ball у клавіатуру.



Рис. 2.70. Координатно-вказівний пристрій з обертальною кулькою (track-ball)



Рис. 2.71. Сенсорна панель

У переносних ПЕОМ як координатно-вказівний пристрій використовують сенсорні панелі (рис. 2.71), в яких положення вказівника на екрані дисплея залежить від положення пальця на сенсорній панелі.

### 2.1.10.3. Ігровий контролер

Ігровий контролер (Game port) відслідковує нахили ручки ігрового маніпулятора (joystick) у двох площинах та натискання кнопок на ньому. Відслідковування нахилів ручки здійснюється шляхом вимірювання опору змінних резисторів, рухомі контакти змінних резисторів зв'язані з ручкою ігрового маніпулятора. До ігрового контролера можна приєднати два маніпулятори. Деякі ігрові маніпулятори доволі реалістично відтворюють органи керування технічних пристроїв – кермо мотоцикла або автомобіля, штурвал літака і т. ін. Їх використовують для побудови різноманітних тренажерів. На рис. 2.69 вказано роз'єднання ігрового порта, на рис. 2.72,а – схема приєднання маніпуляторів до ігрового порта, а на рис. 2.72,б – різні маніпулятори.

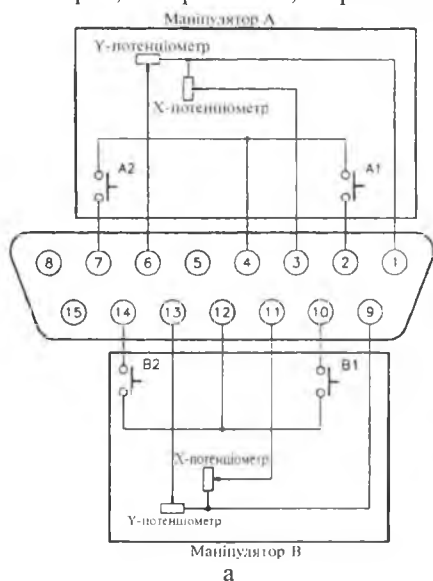


Рис. 2.72. Схема приєднання маніпуляторів до ігрового порта (а) та ігрові маніпулятори (б)

### 2.1.10.4. Паралельні інтерфейси

Паралельні інтерфейси використовують для одночасного введення в ПЕОМ або виведення з ПЕОМ певної кількості біт інформації. Є кілька стандартів паралельних інтерфейсів.

#### 2.1.10.4.1. Інтерфейс Centronics

Паралельний інтерфейс Centronics був розроблений фірмою IBM для приєднання до IBM PC пристроїв друку. Інформація про символ, який потрібно надрукувати, передається по восьми лініях одним байтом (вісім бітів паралельно – паралельний інтерфейс), окрім інформації про символи на пристрій друку передаються команди керування, а з пристрою друку – інформація про стан пристрою друку (немає паперу, триває друк документа і т. ін.). Для приєднання до паралельного інтерфейсу використовують стандартний кабель, на рис. 2.73 зображені роз'єкти цього кабелю з боку системного блоку та з боку пристрою друку, на рис. 2.74 зображені роз'єкти системної плати стандарту ATX, а на рис. 2.75 структура інтерфейсу Centronics.

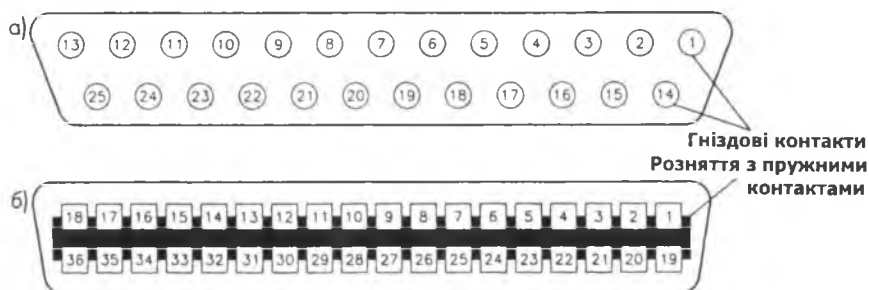


Рис. 2.73. Роз'єкти кабелю інтерфейсу Centronics:  
а – до системного блоку; б – до пристрою друку

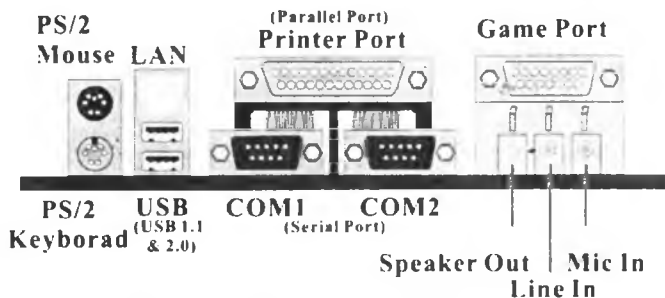


Рис. 2.74. Задня стінка системного блоку з роз'єктами

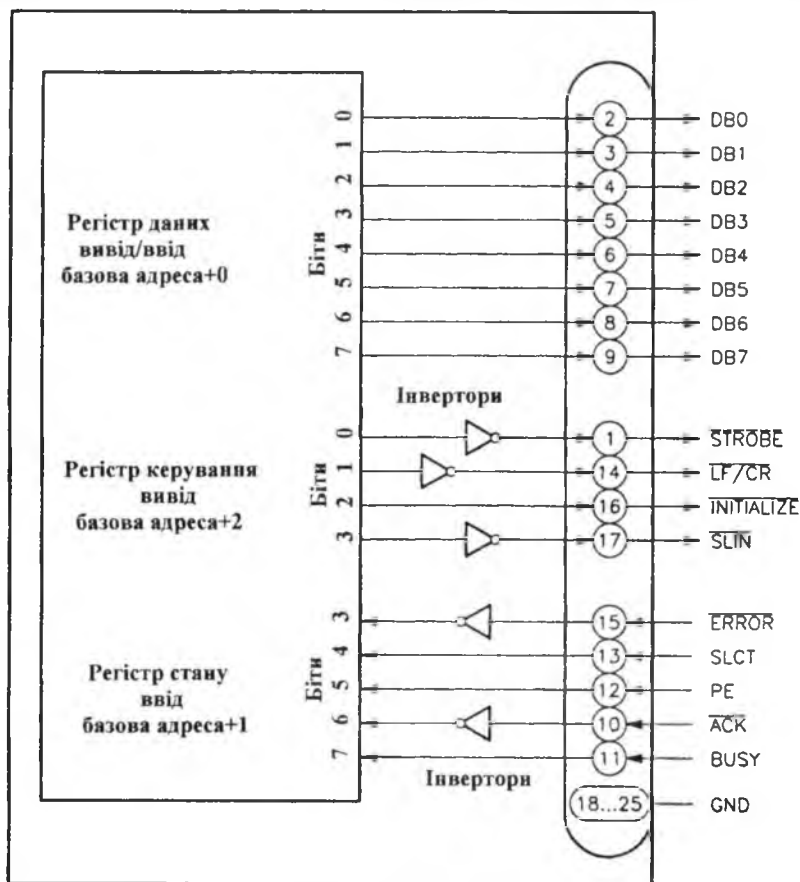


Рис. 2.75. Структура інтерфейсу Centronics

У сучасних системних платах паралельний інтерфейс може працювати у трьох режимах:

**SPP** (Standard Parallel Port) – стандартний паралельний порт, швидкість виводу інформації 100-150 Кб/с;

**EPP** (Enhanced Parallel Port) – покращений паралельний порт, швидкість виводу інформації 1,5-2 Мб/с, можливе приєднання до 64 пристроїв;

**ECP** (Extended Capability Parallel Port) – паралельний порт з розширеними можливостями, швидкість виводу інформації до 4 Мб/с, можливе приєднання до 128 пристроїв.

Режим, у якому працюватиме ваш паралельний порт, визначається пристроєм друку, який ви використовуєте. Задання режиму паралельного порта здійснюється за допомогою програми Setup. Для використання режимів EPP, ECP слід придбати спеціальний (більш дорогий кабель). У деяких випадках паралельний порт використовують для приєднання інших пристроїв (сканерів, пристроїв архівування інформації і т. ін.).

#### 2.1.10.4.2 Інтерфейс SCSI (Small Computer System Interface)

Інтерфейс SCSI в IBM-подібних ПЕОМ у більшості випадків реалізують за допомогою відповідної плати розширення (трапляються системні плати з вбудованим контролером SCSI) і використовують для приєднання жорстких дисків, пристроїв архівування інформації, сканерів і інших пристроїв введення/виведення, які потребують високих швидкостей обміну інформацією. SCSI-пристрої з'єднують у ланцюжок за допомогою плоских кабелів, кожен пристрій повинен мати оригінальний номер (ID), на крайніх пристроях потрібно встановлювати термінатори. Роботу SCSI-пристроїв забезпечують відповідними драйверами, які потрібно інсталиувати при встановленні SCSI-контролера і SCSI-пристроїв. На рис. 2.76 подана схема з'єднань внутрішніх і зовнішніх SCSI-пристроїв, а на рис. 2.77 – термінатор SCSI.

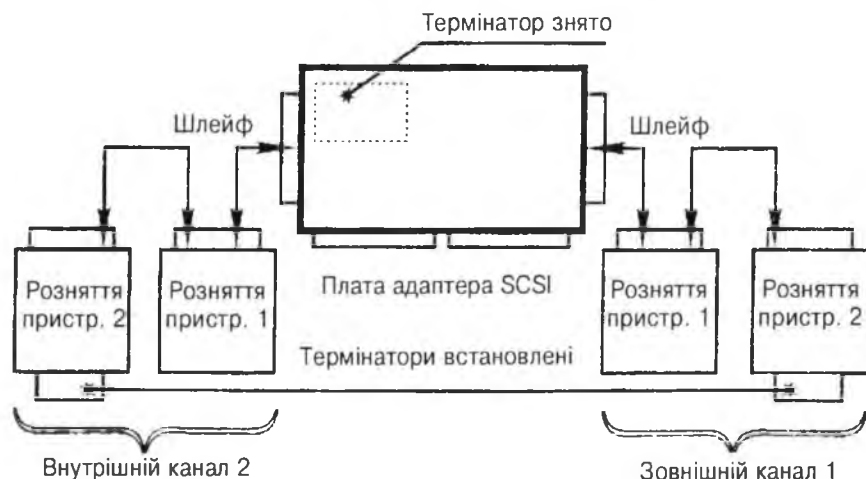


Рис. 2.76. Схема з'єднань внутрішніх і зовнішніх SCSI-пристроїв



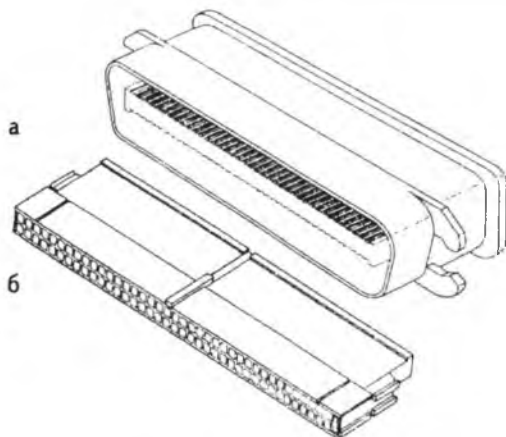


Рис. 2.77. Термінатор SCSI: а – роз'єр SCSI-пристрою; б – термінатор

За двадцять років від створення стандарту SCSI пропускна здатність інтерфейсу зросла від 5 Мб/с до 640 Мб/с (останні версії інтерфейсу – **Ultra160 SCSI**, **Ultra320 SCSI**, **Ultra640 SCSI**). Збільшення щільності розташування складових у корпусах серверів (SCSI в основному використовують при побудові серверів) створило потребу в розробці послідовної версії SCSI з тоншим кабелем. У стадії розробки перебуває три специфікації послідовного SCSI:

**Serial Attached SCSI150** з пропускною здатністю 150 Кб/с;

**Serial Attached SCSI300** з пропускною здатністю 300 Кб/с;

**Serial Attached SCSI600** з пропускною здатністю 600 Кб/с.

#### 2.1.10.5. Послідовні інтерфейси

На відміну від паралельних інтерфейсів, у послідовних інтерфейсах інформація передається *послідовно біт за бітом*. Використання послідовної передачі інформації дозволяє зменшити кількість провідників у кабелі, але ускладнює архітектуру контролера інтерфейсу. Існує кілька стандартів послідовних інтерфейсів.

##### 2.1.10.5.1. Інтерфейс RS-232

Послідовний інтерфейс **RS-232** використовують для маніпуляторів миша, модемів, часом пристроїв друку (застаріло). Окрім інформаційних бітів (один на передачу, другий на приймання інформації) стандартом **RS-232** передбачено використання службових сигналів (запит дозволу

на передавання даних, дозвіл на передавання даних, готовність до приймання інформації, готовність до передавання інформації і т. ін.). Обслуговування пристроїв, приєднаних до послідовного інтерфейсу, здійснюється відповідними програмами (драйверами). Драйвери деяких пристроїв стали стандартними для сучасних операційних систем (маніпулятор миша), однак часом для роботи пристрою з послідовним інтерфейсом потрібно мати інсталяційну дискету або інсталяційний оптичний диск. Приєднання пристроїв до послідовного інтерфейсу **RS-232** здійснюється роз'язками (**Com1**, **Com2**), які вказані на рис. 2.74.

Інтерфейс **RS-232** забезпечує швидкість передавання інформації від 50 біт/с до 115200 біт/с. Швидкість передавання даних визначається обладнанням та лінією передавання даних, які використовуються. На рис. 2.78 подано з'єднання системного блоку і периферійного пристрою інтерфейсом **RS-232**, а на рис. 2.79 – 9- і 25-контактні роз'язки цього інтерфейсу.

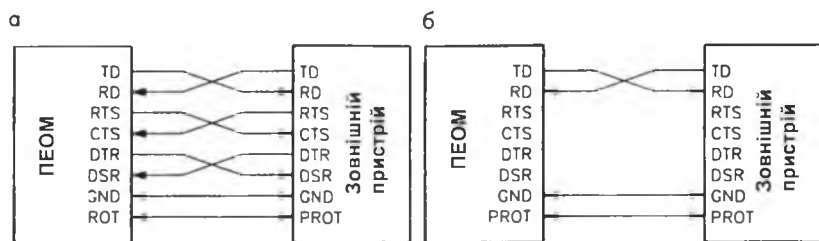


Рис. 2.78. З'єднання системного блоку і периферійного пристрою інтерфейсом **RS-232**:

а – за допомогою нуль-модемного кабелю; б – 4-дротове з'єднання:

**Transmit Data** – лінія передавання інформації; **Receive Data** – лінія приймання інформації; **GND** – лінія заземлення; **PROT** – лінія захисного заземлення;

**RTS, CTS, DTR, DSR** – службові (Hand Shake) сигнали

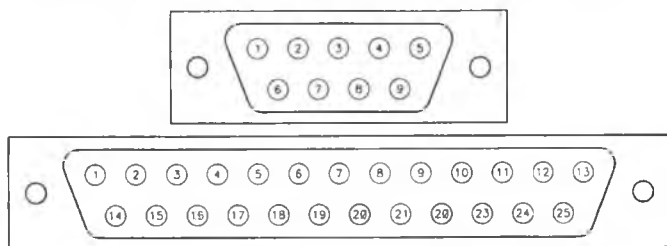


Рис. 2.79. 9- і 25-контактні роз'язки інтерфейсу **RS-232**

### 2.1.10.5.2 Інтерфейс USB

**USB** (Universal Serial Bus – універсальна послідовна шина) призначена для приєднання до системного блока периферійних пристроїв (сканерів, пристроїв друку, клавіатури, мишки, цифрових голосників і т. ін.). Сучасні системні плати мають кілька (до восьми) роз'ємів для приєднання цього інтерфейсу. Оскільки по кабелю **USB** передається напруга живлення 5В, то живлення **USB**-пристроїв може здійснюватися як по кабелю, так і від власних блоків живлення. Пристрої, які забезпечують додаткові точки приєднання, називають розгалужувачами (**Hub**). Пристрій з однією точкою приєднання називають функцією. Використовуючи розгалужувачі і функції, можна будувати деревоподібні структури **USB**-пристроїв. На рис. 2.80 поданий приклад такої структури, а на рис. 2.81 – з'єднувальний кабель **USB**.



Рис. 2.80. Структура з'єднань пристроїв з використанням інтерфейсу **USB**



Рис. 2.81. З'єднувальний кабель **USB**:

А – до системного блоку; В – до периферійного пристрою

Інтерфейс підтримує три швидкості передавання інформації:

- специфікація **USB1.1**:
  - низька 1,5 Мбіт/с;
  - висока 12 Мбіт/с;
- специфікація **USB2.0** – 480 Мбіт/с.

Специфікація **USB 2.0** прийнята у квітні 2000 р. Важливою особливістю специфікації **USB 2.0** є можливість одночасної роботи пристроїв специфікацій 1.1 і 2.0.

Наявність **USB**-портів у всіх видах ПЕОМ – настільних системах, переносних, планшетних і кишенькових ПК привела до появи різноманітних перехідних пристроїв для **USB**:

- перехідник **USB – Centronics**;
- перехідник **USB – PS/2**;
- перехідник **USB – Com**;
- перехідник **USB – SCSI**;
- перехідник **USB – IDE**;
- перехідник **USB – ISA**.

Електронні схеми перехідних пристроїв живляться від інтерфейсу **USB**. Такими перехідниками можуть доповнюватися концентратори **USB**, наприклад **D-Link**.

#### 2.1.10.5.3 Інтерфейс IEEE 1394 (FireWire)

Стандарт послідовного інтерфейсу **IEEE 1394 (FireWire)** був прийнятий у 1995 р. Метою розробки стандарту було створення інтерфейсу, який по швидкості передавання інформації був би співмірний з паралельними інтерфейсами, однак був би дешевшим від них.

Основні властивості **FireWire**:

- можливість роботи з великою кількістю пристроїв (до 63) без використання розгалужувачів (**hub**);
- висока швидкість обміну інформацією – 100, 200, 400 Мбіт/с;
- простота встановлення і використання, пристрої автоматично розпізнаються при їх під'єднанні. Можливе живлення пристроїв від інтерфейсу (струмом до 1,5 А);
- відносно низька ціна.

Робота інтерфейсу **FireWire** у ПЕОМ забезпечується спеціальною платою розширення (деякі виробники системних плат, наприклад **Asus**, пропонують системні плати із вбудованим інтерфейсом **FireWire**).

Інтерфейс дозволяє будувати структуру, яка є поєднанням деревоподібної і ланцюжкової схем, на варіанти приєднання пристроїв до інтерфейсу накладаються такі обмеження:

- між будь-якою парою вузлів повинно бути не більше 16 кабельних сегментів;
- довжина стандартного кабеля не більше 4,5 м.;
- сумарна довжина кабеля має бути не більшою від 72 м.

На рис. 2.82 зображені приклад з'єднання пристроїв з використанням цього інтерфейсу, а на рис. 2.83 – з'єднувальний кабель цього інтерфейсу.

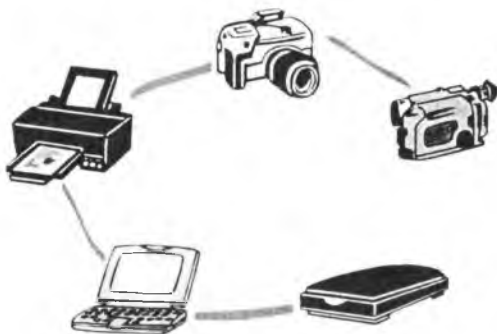


Рис. 2.82. Структура з'єднань пристроїв з використанням інтерфейсу **FireWire**



Рис. 2.83. З'єднувальний кабель інтерфейсу **FireWire**

Нова версія інтерфейсу **IEEE 1394b (FireWire800)** розрахована на такі швидкості передавання інформації – 800, 1600, 3200 Мбіт/с.

#### 2.1.10.5.4. Інтерфейс цифрових музичних інструментів **MIDI**

Цифровий інтерфейс музичних інструментів **MIDI** є двоскерованим послідовним інтерфейсом з швидкістю передавання інформації 31,25 Кбіт/с. Його використовують для приєднання синтезаторів, записуючих і відтворюючих змішувачів сигналів, пристроїв спецефектів та інших електромузичних інструментів. **MIDI**-пристрої приєднують до системного блоку за допомогою роз'єднання, зображеного на рис. 2.69 (ігровий порт).

### 2.1.10.6. Бездротові інтерфейси

Якщо подивитися на ПЕОМ, до якої приєднані голосники, пристрій друку, модем, сканер ззаду, то плутанина кабелів (мережевих і сигнальних) навряд чи Вам сподобається. Для бездротового обміну інформацією між системним блоком та периферійними пристроями використовують бездротові інтерфейси. Розглянемо їх.

#### 2.1.10.6.1. Інфрачервоний інтерфейс (IrDA)

Сучасні системні плати, як правило, мають роз'єкти для приєднання інфрачервоного інтерфейсу. Оскільки інформація передається світловими імпульсами, то такий інтерфейс є бездротовим. Використання **IrDA** можливе після встановлення на системному блоці та периферійному пристрої (клавіатура, мишка, пристрій друку і т. ін.) пар “приймач-передавач” та інсталяції відповідного програмного забезпечення. Існує кілька стандартів **IrDA** з різними швидкостями передавання інформації:

**IrDA SIR** – 9,6-11,2 Кбіт/с;

**Sharp ASK** – 9,6-57,6 Кбіт/с;

**IrDA MIR** – 1,2 Мбіт/с;

**IrDA FIR** – 4 Мбіт/с.

#### 2.1.10.6.2 Інші бездротові інтерфейси

Крім інфрачервоного інтерфейсу для бездротового обміну інформацією використовують радіоінтерфейси – **IEEE 802.11** та **Bluetooth**. Оскільки ці інтерфейси в основному застосовують для бездротового обміну інформацією в локальних мережах, то ми їх розглянемо далі.

### Контрольні запитання

1. Які пристрої і інтерфейси зараховують до системних засобів вводу/виводу?
2. У чому полягає відмінність між паралельними і послідовними інтерфейсами?
3. Перерахуйте паралельні інтерфейси, які використовуються у IBM-подібних ПЕОМ. Поясніть їх призначення.
4. Перерахуйте послідовні інтерфейси, які використовуються у IBM-подібних ПЕОМ. Поясніть їх призначення.
5. На яких фізичних принципах ґрунтується робота клавіатур різних конструкцій?
6. На яких фізичних принципах ґрунтується робота координатно-вказівних пристроїв різних конструкцій?

## 2.2. Особливості ПЕОМ фірми Apple

Перша ПЕОМ Apple-I була створена у 1976 р. Стівом Джобсом і Стефаном Возняком як ігровий пристрій з виведенням зображення на побутовий телевізор. Програмісти, яких С. Джобс і С. Возняк попросили оцінити створене, заявили, що насправді Apple-I є комп'ютером, а не ігровим пристроєм. Вже у 1977 році створена С. Джобсом і С. Возняком фірма Apple розпочала серійне виробництво ПЕОМ Apple-II (рис. 2.84) із роздрібною ціною \$1298. Доволі швидко для Apple-II було створено текстовий редактор (Електричний олівець) і табличний процесор (VisiCalc). Саме наявність цих програм і поміркована ціна Apple-II стали причиною нечуваного для США явища – службовці фірм купували Apple-II за свої кошти для використання на роботі. Усього було продано 1 300 000 ПЕОМ Apple-II. Наступні проекти фірми Apple не були такими вдалим (Apple-III, Lisa), внаслідок чого С. Возняк відійшов від керівництва компанією. У 1984 р. фірма Apple розпочала виробництво лінійки ПЕОМ з назвою Macintosh. ПЕОМ Mac I (роздрібна ціна \$2500) мала такі характеристики:

- мікропроцесор Motorola 68000;
- 128 Кб ОЗПр.;
- 3.5" ГМД.;
- графічний інтерфейс.



а



б

Рис. 2.84. ПЕОМ фірми Apple:  
а – Apple-II; б – Mac I

ПЕОМ фірми Apple будувалися на мікропроцесорах фірми Motorola (згодом до виробництва мікропроцесорів для ПЕОМ Apple, Macintosh приєдналася фірма IBM). У таблиці подано назви мікропроцесорів, які використовувалися фірмою Apple для виробництва своїх ПЕОМ.

Мікропроцесор	Розрядність	Швидкодія (тактова частота) МГц	Назва ПЕОМ
Motorola 68000	16 біт	6-16	Apple, Macintosh
Motorola 68020	32 біт	16-20	Macintosh***** Беста (виробн. СРСР)
Motorola 68030	32 біт	16-50	Macintosh*****
Motorola 68040	32/64 біт	16-60	Macintosh*****
Power PC G1	32/64 біт	33 -120	Power Macintosh,, Power Book
Power PC G2	32/64 біт	До 300	Power Macintosh, iMac, eBook, Power Book
Power PC G3	32/64 біт	До 450	Power Macintosh, iMac, eBook, Power Book
Power PC G4	32/64 біт	До 1000	Power Macintosh, iMac, eBook. Power Book
Power PC G4+	32/64 біт	До 1833	Power Macintosh, iMac, eBook. Power Book
Power PC G5+	32/64 біт	До 2700	Power Macintosh, iMac, eBook, Power Book

Фірма Apple є законодавцем “комп’ютерної моди”, передовий дизайн її ПЕОМ лише згодом стає набутком IBM-подібних ПЕОМ. На рис. 2.85 подано зображення сучасних ПЕОМ фірми Apple.



Рис. 2.85. Сучасні ПЕОМ фірми Apple:  
а – Power Macintosh G5; б – eBook  
(усі складові розташовані в корпусі дисплея); в – iMac



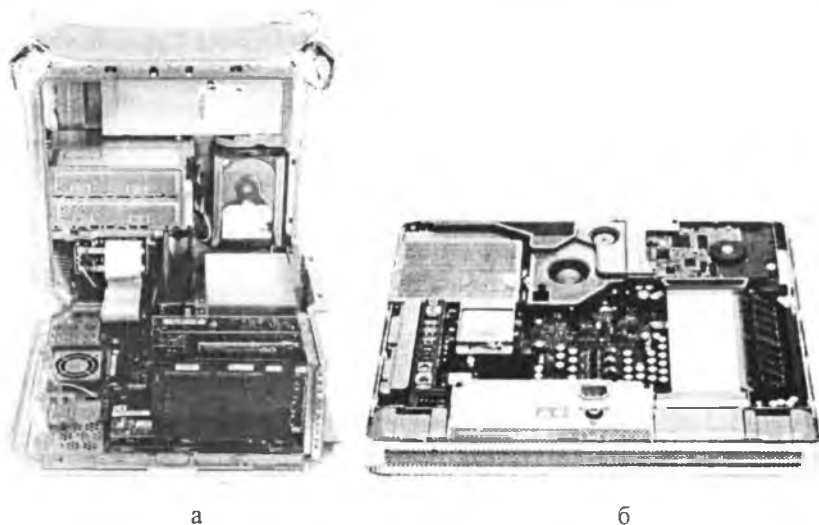


Рис. 2.86. Внутрішня будова ПЕОМ:

а – Power Macintosh; б – iMac G5 [товщина корпусу 5 см.]

Наведемо короткі відомості про мікропроцесори фірми Motorola (серія MC 680 XX), які є базою для побудови комп'ютерів компанії Apple Computer і відомі під назвою Macintosh. Сім'я цих МП містить 16- (68000, 68008, 68012) і 32-розрядні (68020, 68030) процесори: 32-розрядні МП сумісні з 16-розрядними. МП працюють у двох режимах: користувачькому та супервізора. У 32-розрядних МП збільшено кількість режимів сумісності; передбачено масштабування у кількох режимах адресування (1, 2, 4 або 8); додано 16 нових команд ЦП і 7 команд співпроцесора. Блок керування пам'яттю міститься в окремій мікросхемі.

Типи даних, які опрацьовує МП: цілі, біти і двійково-десяткові цифри. Цілі можуть бути відображені у вигляді байтів, слів або довгих слів. Байтові дані можна адресувати по межах парних або непарних адрес, а слова – тільки по парних адресах.

Структура МП така:

- блок регістрової пам'яті:
  - 68000: вісім 32 розрядних регістрів даних;
    - дев'ять 32 розрядних адресних стекових регістрів;
    - програмний лічильник;
    - регістр стану (байт користувача і байт супервізора);

- 68030: шістнадцять 32 розрядних РЗП (даних і адрес);
  - два 32 розрядних вказівники стеку супервізора;
  - блок аналізу особливих ситуацій. Вектори особливих ситуацій розташовані у пам'яті, починаючи з комірки 0 до 1023 (кожний вектор займає 4 байти);
  - блок керування пам'яттю.
- 6800: 12 режимів адресації: регістрова пряма; безпосередня; відносна; абсолютна; неявна; регістрова непряма;
- 68030: 18 режимів адресації. Додана адресація по mod4, яка використовується для швидкого вибирання напівбайтів і сторінок, а також статистичних декодувань стовпчиків;
- блок опрацювання команд. Команди поділяють на вісім груп:
  - пересилання даних;
  - зсуву і циклічного зсуву;
  - опрацювання бітів (розрядів регістра чи бітів пам'яті);
  - логічних операцій;
  - операцій над двійково-десятковими числами;
  - двійкової арифметики;
  - програмного керування;
  - спеціальні (для організації роботи з мовами високого рівня). Наприклад, передавання параметрів, перевірка сторінок, організація часто повторюваних циклів та ін.
- У наступних моделях (20 і 30):
  - блок спряження із співпроцесором;
  - трирівневий конвеєр команд;
  - механізм динамічної зміни розрядності шини.
- У МП 68030:
  - засоби паралельного виконання операцій;
  - кеш даних і команд (по 256 байт кожний);
  - пристрій сторінкової організації пам'яті;
  - кеш команд.

Найновішим і найпотужнішим мікропроцесором у сімействі Power PC, на яких фірма Apple понад 10 років будує свої ПЕОМ, є мікропроцесор Power PC G5. На рис. 2.87 – 2.90 подані спрощена блок-схема цього мікропроцесора, структура ПЕОМ Power Mac G5, стандартні інтерфейси цієї ПЕОМ і компонування її системного блоку.



Рис. 2.87. Блок-схема мікропроцесора Power PC G5

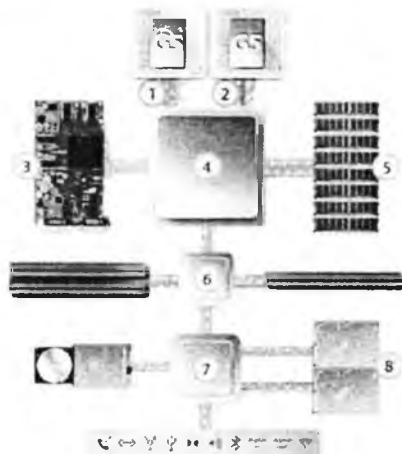


Рис. 2.88. Структура ПЕОМ Power Mac G5:

1, 2 – локальна шина мікропроцесора (FSB) з частотою 1 ГГц (окремо на кожний Power PC G5); 3 – графічний контролер AGP 8X; 4 – швидка мікросхема системної логіки для роботи з мікропроцесорами; 5 – двоканальна оперативна пам'ять DDR із частотою шини 400 МГц.; 6 – мікросхема системної логіки для забезпечення роботи 64-бітної шини PCI-X; 7 – мікросхема системної логіки для забезпечення роботи дискових пристроїв і стандартних інтерфейсів

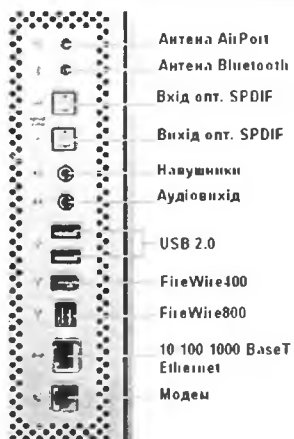


Рис. 2.89. Стандартні інтерфейси Power Mac G5

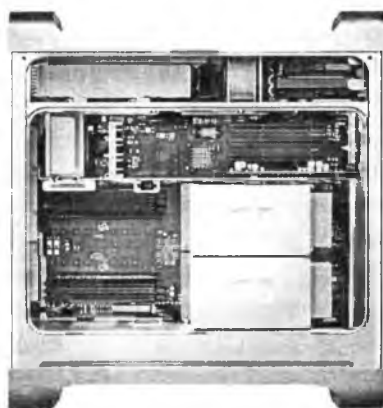


Рис. 2.90. Компонування системного блоку Power Mac G5

### Контрольні запитання

1. Чому ПЕОМ фірми Apple мають менше поширення в Україні у порівнянні з IBM-подібними ПЕОМ?
2. На яких мікропроцесорах будують ПЕОМ фірми Apple?
3. Які конструктиви використовує фірма Apple для своїх ПЕОМ?
4. У якому році розпочалося серійне виробництво Apple-II?
5. У якому році розпочалося серійне виробництво IBM PC?

## 2.3. Особливості портативних ПЕОМ

Портативні ПЕОМ сьогодні поділяються на чотири групи:

- переносні ПЕОМ;
- кишенькові ПЕОМ;
- планшетні ПЕОМ;
- комунікатори.

Розглянемо їх докладніше.

### 2.3.1. Переносні ПЕОМ

Загалом структура переносної ПЕОМ дуже подібна до структури настільної ПЕОМ, однак жорсткі обмеження на розмір і вагу пристрою змушують використовувати дещо інші технічні рішення. Корпуси переносних ПЕОМ виготовляють з дуже міцної пластмаси або алюмінієво-магнієвого сплаву, дисплей обов'язково плоский (TFT) і вбудований у відкидну кришку, клавіатура зменшених розмірів, графічний координатно-вказівний пристрій вбудований у корпус (сенсорна панель або trackball [застаріло]), для живлення електронних схем ПЕОМ використовують блок живлення від електромережі 220В (110В) та акумулятори для автономної роботи ПЕОМ протягом кількох годин. Пристрої читання інформації з гнучких магнітних дисків та оптичних дисків мають зменшену висоту і глибину, а жорсткі магнітні диски зменшені розміри. У деяких



Рис. 2.91. Переносна ПЕОМ

переносних ПЕОМ використовують відсік для встановлення змінних пристроїв для запису/читання інформації з ГМД, або читання (запису/читання) інформації з оптичних дисків. Використання змінних пристроїв дозволяє зменшити габарити і вагу переносної ПЕОМ, але ускладнює користування ним. Використання зовнішніх пристроїв запису/читання інформації з ГМД та читання (запису/читання) інформації з оптичних дисків також дозволяє зменшити габарити і вагу переносної ПЕОМ, однак не всім користувачам ПЕОМ це подобається. На рис.2.91 зображено переносну ПЕОМ.

Для приєднання зовнішніх пристроїв використовують стандартні інтерфейси, роз'яття цих інтерфейсів виведені на задню стінку переносної ПЕОМ.

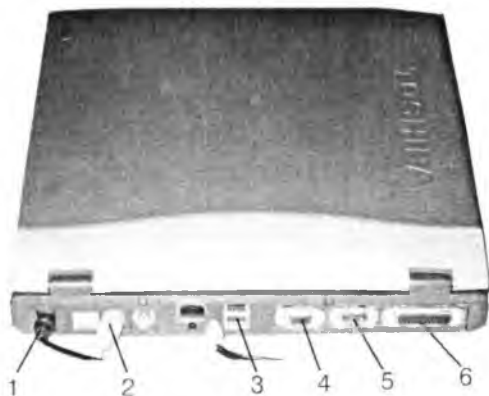


Рис. 2.92. Роз'яття для приєднання зовнішніх пристроїв:

- 1 – роз'яття для зовнішнього блоку живлення; 2 – порт PS/2 для мишки;
- 3 – роз'яття інтерфейсу USB; 4 – роз'яття для дисплея або проектора;
- 5 – роз'яття інтерфейсу COM; 6 – роз'яття інтерфейсу Centronics

До переносної ПЕОМ ви можете під'єднати зовнішню мишку, проектор або дисплей, пристрій друку, сканер, пристрої архівування інформації і т. ін.

Для приєднання додаткових пристроїв у переносних ПЕОМ використовують також шину **PCMCIA**, введену Personal Computer Card Memory Internation Association. Пристрої стандарту **PCMCIA** мають стандартні розміри і встановлюються у спеціальний відсік у корпусі переносної ПЕОМ.

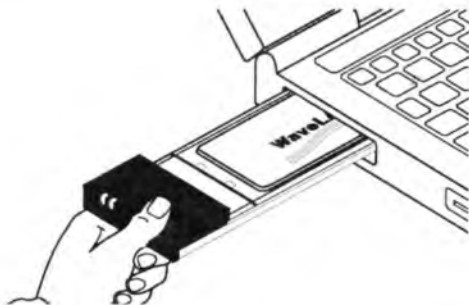


Рис. 2.93. Встановлення плати стандарту PCMCIA

Усі пристрої для переносних ПЕОМ виготовляють з мінімальним енергоспоживанням для забезпечення якомога довшого сеансу роботи ПЕОМ з живленням від акумуляторів. Ціни на переносні ПЕОМ у 2-5 разів вищі від цін на співмірні за потужністю стаціонарні ПЕОМ.

Окрім, так би мовити, “класичних” переносних ПЕОМ (notebook) з шириною корпусу більше 300 мм (для моделей з 15” дисплеєм ширина корпусу становить 320-335 мм) виробники пропонують переносні ПЕОМ зменшених розмірів (subnotebook), які мають класичну будову і розміри, співмірні з кишеньковими ПЕОМ. Діагональ дисплеїв таких ПЕОМ зменшена (не більше 10”), а внаслідок обмежень на товщину і масу корпусу такі ПЕОМ не мають вбудованих пристроїв читання інформації з ГМД і оптичних дисків.

### 2.3.2. Кишенькові ПЕОМ

Ще меншими від subnotebook є кишенькові ПЕОМ. Завдяки зв’язку через інфрачервоний інтерфейс або **Bluetooth** з мобільним телефоном користувачі кишенькових ПЕОМ отримують доступ до файл-серверів і Інтернету. Для зв’язку зі стаціонарними ПЕОМ використовують **RS-232**, **USB**, **Bluetooth** або інфрачервоний інтерфейс.

Програмне забезпечення кишенькових ПЕОМ складається з:

- операційної системи – **Palm OS**, **Microsoft Windows CE**, **EP0C32**, **Linux**;
- прикладних програм – календар, калькулятор, поштова програма, ігри, програми перегляду електронних книг, текстових документів, електронних таблиць.

Усі кишенькові ПЕОМ обладнані сенсорними дисплеями з функцією рукописного вводу. До них можна під’єднати клавіатуру (складану або гнучку), деякі моделі мають вбудовану клавіатуру зменшених розмірів. На рис. 2.94 подані зображення кишенькових ПЕОМ, на рис. 2.95 – змінні пристрої для кишенькових ПЕОМ, а на рис. 2.96 – кишенькова ПЕОМ **Palm VX** із розкладною клавіатурою.



Рис. 2.94. Кишенькові ПЕОМ:

- а – NX70V Color Clie (кольоровий сенсорний дисплей 320х480 крапок, оперативна пам'ять 16Мб, цифрова фотокамера, програвач MP3, Palm OS);  
 б – Sharp Zaurus (кольоровий сенсорний дисплей 240х320 крапок, оперативна пам'ять 64Мб, вбудована клавіатура, Linux)



Рис. 2.95. Змінні пристрої для кишенькових ПЕОМ:

- а – NX70V Color Clie із флеш-картою і цифровою фотокамерою у форматі флеш-карти; б – адаптер GPS; в – клавіатура для iPaq Compaq





Рис. 2.96. Кишенькова ПЕОМ Palm VX із розкладною клавіатурою

Якщо в переносних ЕОМ використовують мобільні версії мікропроцесорів для стаціонарних ПЕОМ (Intel Pentium M, Intel Celeron M, AMD Duron M, AMD Sempron M), то в кишенькових ПЕОМ використовують спеціалізовані мікропроцесори зі зменшеним енергоспоживанням. Як приклад розглянемо мікропроцесор Intel PXA250.

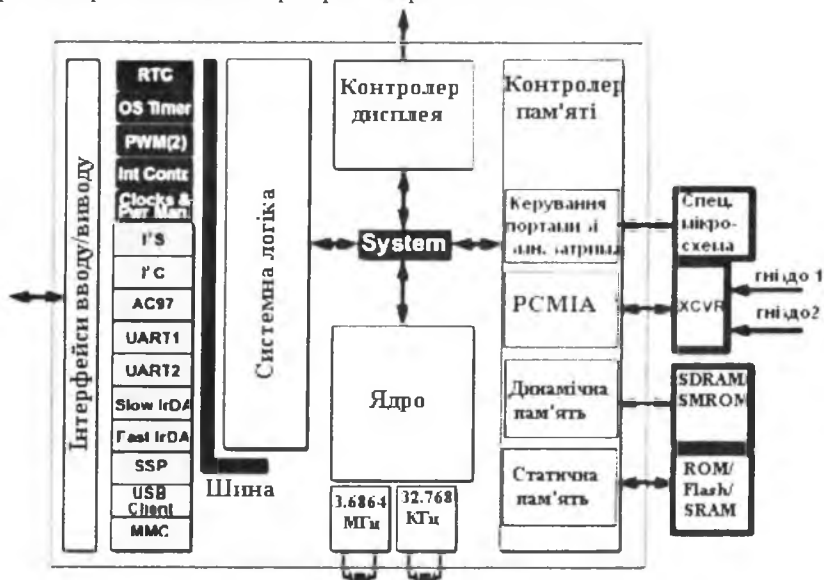


Рис. 2.97. Мікропроцесор Intel PXA250

Процесор Intel PXA250 базується на 32-бітному RISC-ядрі Intel Xscale із суперскалярним виконанням команд. Ядро XScale процесора Intel PXA250 має такі характеристики:

- 32-Кб кеш команд;
- 32-Кб кеш даних;
- 2-Кб міні-кеш даних (окремо від L1);
- 2-Кб міні-кеш команд (для резидентної програми налагодження);
- 40-бітний акумулятор і розширений 16-розрядний пристрій множення для опрацювання мультимедійної інформації;
- засоби керування пам'яттю інструкцій і даних та ін.

Окрім ядра в процесор інтегровані контролер пам'яті (динамічна і статична пам'ять), графічний контролер, контролер системних засобів вводу/виводу (I2S, I2C, AC'97, UART1, UART2, Slow IrDA, Fast IrDA, USB). Процесор Intel PXA250 підтримує роботу ядра з тактовою частотою 100–400 МГц. Різні режими експлуатації забезпечують масштабування системи як за продуктивністю, так і за енергоспоживанням (чим вища напруга живлення [в межах допустимих значень], тим вище значення тактової частоти, і навпаки – чим менша напруга живлення, тим нижче максимальне значення тактової частоти). Максимальне енергоспоживання процесора Intel PXA250 складає 1,4 Вт, мінімальне – менше 500 мВт. На рис. 2.98 подано приклад під'єднання компонент кишенькової ПЕОМ до процесора Intel PXA250, а на рис. 2.99 – структура кишенькової ПЕОМ Pocket LOOX 600.

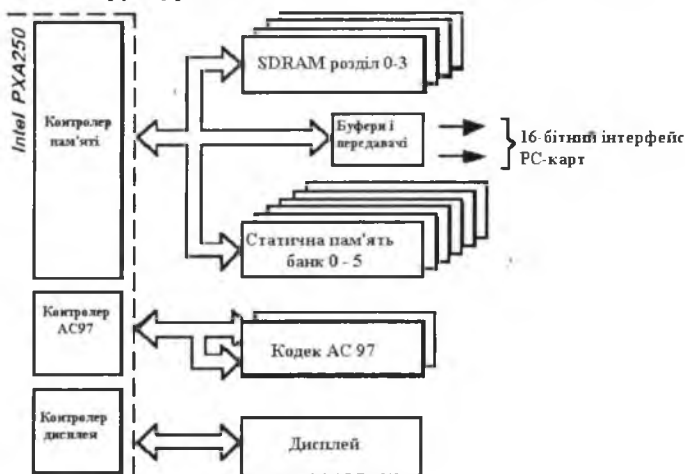


Рис. 2.98. Під'єднання компонент кишенькової ПЕОМ до процесора Intel PXA250

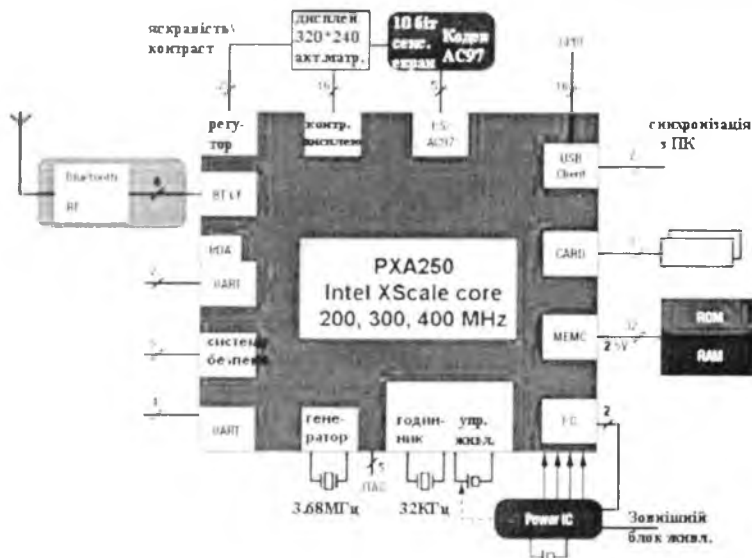


Рис. 2.99. Структура кишенькової ПЕОМ Pocket LOOX 600

### 2.3.3. Планшетні ПЕОМ

Удосконалення технологій сенсорних екранів, голосового вводу, бездротового зв'язку спричинили розроблення планшетних ПЕОМ (Tablet PC) – такий пристрій при розмірах грубого журналу повинен мати потужність сучасної ПЕОМ. Розробки планшетних ПЕОМ ведуться кілька останніх років. На рис. 2.100 подано ViewSonic Tablet PC V1.100 з такими характеристиками:

- процесор Pentium III 866 МГц;
- оперативна пам'ять 256Мб;
- жорсткий магнітний диск 20Гб;
- сенсорний дисплей з діагоналлю 10,4";
- операційна система Microsoft Windows Tablet PC Edition.



Рис. 2.100. Планшетна ПЕОМ зі з'єднувальною станцією

### 2.3.4. Комуникатори

Поєднання в одному пристрої кишенькової ПЕОМ і мобільного телефону привело до появи комуникаторів – ви можете розмовляти по телефону або працювати з цим пристроєм як з кишеньковою ПЕОМ. На рис. 2.101 подано зображення комуникаторів.



Рис. 2.101. Комуникатори:

- а) Nokia 9210 (мобільний телефон з протилежного боку комуникатора);  
б) aI-100RPNokia 9210 (Java-процесор)

#### Контрольні запитання

1. Перерахуйте типи портативних ПЕОМ і назвіть галузі їх застосування.
2. Які дисплеї використовують у портативних ПЕОМ?
3. Які системні інтерфейси використовують у портативних ПЕОМ?
4. Чи є можливість використання плат розширення у портативних ПЕОМ?
5. Які запам'ятовуючі пристрої використовують у портативних ПЕОМ?

## 2.4. Робочі станції і промислові ПЕОМ

Ще до появи персональних ЕОМ різні виробники продукували ЕОМ, які призначалися для використання одним користувачем – Промінь, Мир-2, PDP-8, LSI-11, microVAX, які з часом отримали назву мікро-ЕОМ. На відміну від персональних ЕОМ, мікро-ЕОМ були орієнтовані на використання професіоналами. Особливе місце серед професійних мікро-ЕОМ посідають робочі станції – ЕОМ з потужним 32- або 64-розрядним мікропроцесором, великою оперативною пам'яттю і потужною графічною підсистемою. Слід зауважити, що перша операційна система з графічним інтерфейсом була реалізована у 1982 р. у робочій станції

Xerox Star. Суттєво менші обсяги виробництва робочих станцій, значно вищі їх технічні характеристики зумовлюють набагато вищі ціни на робочі станції у порівнянні з цінами на персональні ЕОМ. Розглянемо особливості архітектури персональних станцій на прикладі сімейства Silicon Graphics. Насамперед слід наголосити на тому, що архітектура робочої станції сильно залежить від її призначення. На рис. 2.102 подана архітектура Visual Workstation – робочої станції для роботи з дво- і тривимірною графікою. Для уникнення копіювання інформації з оперативної пам'яті у графічну або аудіо-пам'ять у Visual Workstation використовують спільну системну пам'ять (кілька гігабайт), яка розподіляється між підсистемами робочої станції. Робоча станція Visual Workstation базується на наборі системної логіки Cobalt™ з наступними показниками:

- швидкість обміну інформацією між оперативною пам'яттю і графічною підсистемою – 3,2 Гб/с.;
- швидкість обміну інформацією між оперативною пам'яттю і системою вводу/виводу – 1,6 Гб/с.;
- швидкість обміну інформацією між оперативною пам'яттю і процесором – 800 Мб/с.;
- виконання функцій апаратного прискорювача дво- і тривимірної графіки у наборі системної логіки Cobalt™;
- апаратна реалізація операцій затінення, освітленості і промальовування моделей.

Очевидно, що з часом такі показники стають доступними і для IBM-подібних ПЕОМ.



Рис. 2.102. Архітектура IVC (Integrated Visual Computing)

На рис. 2.103 подана архітектура UMA, на якій базуються робочі станції O2. Як і в архітектурі IVC, для всіх підсистем робочої станції у ній використовують спільну оперативну пам'ять зі швидкістю доступу 2,1 Гб/с.



Рис. 2.103. Архітектура UMA (Unified Memory Architecture)

Для побудови робочих станцій сімейства Silicon Graphics використовують мікропроцесори Intel Xeon і Rxx000 (R12000-R8000). У таблиці подані технічні характеристики сучасних робочих станцій Silicon Graphics, а на рис. 2.104-105 – зображення робочих станцій.

	Silicon Graphics® Fuel	Silicon Graphics® Tezro™	Onyx4 UltimateVision
Процесор	MIPS R16000A, $f_T$ – 700/800 MHz, 4MB L2 cache; 200 MHz front-side bus	1, 2 або 4 MIPS R16000A, $f_T$ – 700/800 MHz, 4MB L2 cache; 200 MHz front-side bus	2 – 64 MIPS R16000A, $f_T$ – 700/800 MHz, 4MB L2 cache; 200 MHz front-side bus, 2–32 графічних процесори
Опера- тивна пам'ять	512MB–4GB (DDR SDRAM)	512MB–16GB (DDR SDRAM)	16ГБ–128 GB (DDR SDRAM)
Сфери застосу- вання	Medical Imaging Scientific Visualization 3D Animation Oil and Gas (seismic interpretation) Visual Simulation Geospatial imaging	MCAD/MCAE Digital mockup (DMU) MDO Team room visualization Scientific visualization Data fusion: command-and- control centers Oil and gas (seismic interpretation and reservoir) Visual simulation Editing and compositing Film restoration and mastering HD acquisition Geospatial imaging Medical imaging	MCAD/MCAE Digital mockup (DMU) MDO Team room visualization Scientific visualization Data fusion: command-and- control centers Oil and gas (seismic interpretation and reservoir) Visual simulation Editing and compositing Film restoration and mastering HD acquisition Geospatial imaging Medical imaging



Рис. 2.104. Робоча станція Silicon Graphics Fuel

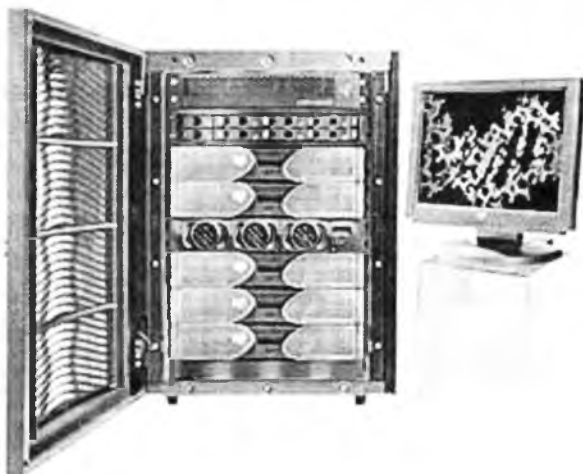


Рис. 2.105. Робоча станція Silicon Graphics Onyx4

Потреба у системах управління технологічними процесами спричинила появу промислових ПЕОМ (київський Інститут кібернетики розробляв спеціалізовані ЕОМ управління технологічними процесами, наприклад, ЕОМ Днепр. Однак ціна таких ЕОМ була доволі високою). Особливі умови виробничих приміщень (вібрації, запиленість і т. ін.) вимагають використання дещо інших конструктивних рішень (амортизаційні підвіски ЖМД, фільтрація потоку повітря і т. ін.). На рис. 2.106 зображено промислову ПЕОМ E.VEREST T9.



Рис. 2.106. Переносна промислова ПЕОМ

## 2.5. Кластери

Одним із методів побудови багатопроцесорних систем є використання серійних ПЕОМ зі швидкісними каналами зв'язку між ними. Такий підхід є менш вартісним, аніж побудова спеціально спроектованої багатопроцесорної системи (Deep Blue, Earth Simulator, Cray). Одним із перших проєктів по створенню кластера був проєкт Beowulf (1994 р.). Цей кластер складався з 16 ПЕОМ і мав продуктивність 70 Mflops (flops – floating-point operations per second – операцій з плаваючою комою за секунду), вартість – \$40 000. Можна виділити два напрямки використання кластерів:

- обчислювальні кластери;
- серверні застосування (див. далі).

Обчислювальні кластери в Україні працюють у Київському національному університеті імені Т. Г. Шевченка, Львівському інституті фізики конденсованих систем НАН України, Інституті кібернетики ім. В.М. Глушкова НАН України. Окрім обчислювальних кластерів великі корпорації використовують кластери як сервери (Київстар GSM, UMC, Промінвестбанк, Укресімбанк, Міністерство екології і природних ресурсів, ВАТ ЛисичанськНафтаОргСинтез, Іллічівський морський порт [перелік неповний]). На рис. 2.107 подано фотографію найпотужнішого обчислювального кластера в Україні, який був уведений в дію у 2004 р.





Рис. 2.107. Кластер Інституту кібернетики ім. В. М. Глушкова НАН України

У таблиці подано технічні характеристики кластерів, які встановлені в Україні.

Організація	Рік встановлення	Кількість процесорів	Швидкість обміну інформацією	Продуктивність	Програмне забезпечення
Інформаційно-обчислювальний центр Київського національного університету імені Тараса Шевченка	1999	34 (14 Pentium III 1ГГц, 512 Мб ОЗП + 16 Intel Xeon 2.4ГГц, 512 Мб ОЗП + 4 Intel Xeon 3.2 ГГц, 2048 Мб ОЗП)	1 Гбіт/с.	Пікова 120 Gflops, Linpack 32,5 Gflops	Fedora Linux – 1, CHPOX, torque, maui LAM, MPICH, PVM, прикладні програми
Львівський Інститут фізики конденсованих систем НАН України	2002	12 (2 Athlon MP 2200+, 2Гб ОЗП)	Fast Ethernet ~90Mbit/s Gigabit Ethernet ~650Mbit/s	17.5 Gflops на 8 вузлах +Fast Ethernet	Red Hat Linux 7.1, Mosix, OpenPBS MPICH 1.2.4
Інститут кібернетики ім. В.М.Глушкова НАН України	2004	32 Intel Xeon	немає інформації	немає інформації	немає інформації
Інститут кібернетики ім. В.М.Глушкова НАН України	Планується у 2005	64 Intel Itanium	немає інформації	немає інформації	немає інформації

Окрім централізованих кластерних систем, які складаються з певної кількості ПЕОМ, що встановлюються в одному приміщенні, використовують децентралізовані кластерні системи. Децентралізовані кластерні системи складаються з ПЕОМ, які розташовані у різних приміщеннях і об'єднані локальною мережею. Для керування роботою кластера потрібне відповідне програмне забезпечення (операційна система і система керування задачами). Подальшим розвитком децентралізованих кластерних систем можна вважати системи з розподіленими обчисленнями, які складаються із ПЕОМ, розташованих у різних містах, країнах, континентах і об'єднаних глобальною мережею Інтернет (проекти FightAIDSatHome, SETI@home).

### Контрольні запитання

1. З якою метою створюють кластери?
2. Які конструктиви використовують при побудові кластерів?
3. Що таке децентралізовані кластерні системи?
4. Коли був створений перший кластер?
5. Перерахуйте відомі вам українські кластери.

## 2.6. Серверні ЕОМ

Особливі завдання, які покладаються на серверні ЕОМ, сприяли розробці особливих конструктивів серверів. Потреба у якомога швидшому опрацюванні запитів від багатьох ПЕОМ у мережі викликала появу багатопроцесорних архітектур і побудову дискової підсистеми на основі кількох ЖМД з особливим розташуванням інформації на них (RAID-масиви).

Залежно від завдань, які покладаються на сервер, їх можна розділити на сервери:

- початкового рівня (файл-сервер, Інтернет-сервер, термінальний сервер, сервер баз даних для малих та середніх робочих груп [від кількох ПЕОМ до кількох десятків ПЕОМ]);
- середнього рівня (файл-сервер, Інтернет-сервер, термінальний сервер, сервер баз даних, сервер програмного забезпечення для корпорації [від кількох десятків ПЕОМ до кількох сотень ПЕОМ]);

- верхнього рівня (сервер розподілених баз даних, центральний корпоративний термінальний сервер, SQL-сервер, Exchange-сервер).

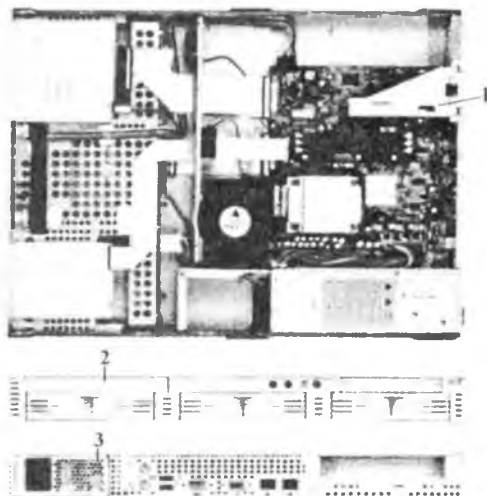


Рис. 2.108. П'єдестальний варіант сервера Asus AP140R  
(розміри – 595мм x 430мм x 44мм):

1 – вид згори із знятою кришкою, 2 – вид спереду, 3 – вид ззаду

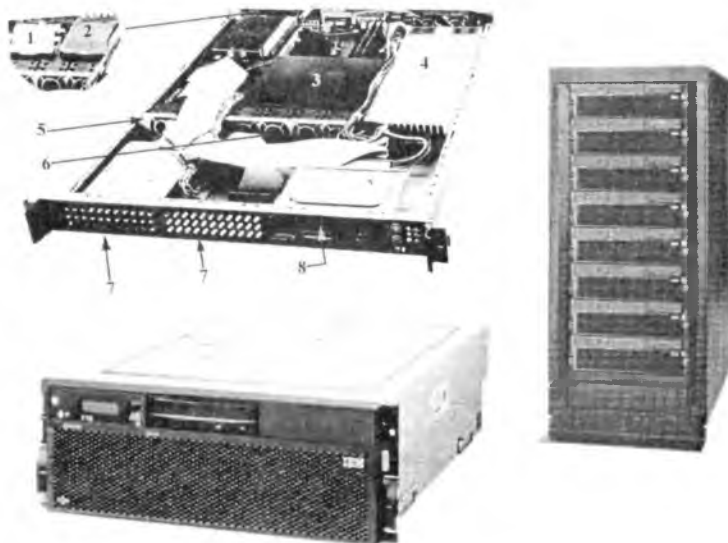


Рис. 2.109. Стійкові варіанти сервера (зліва модулі, справа стійка)

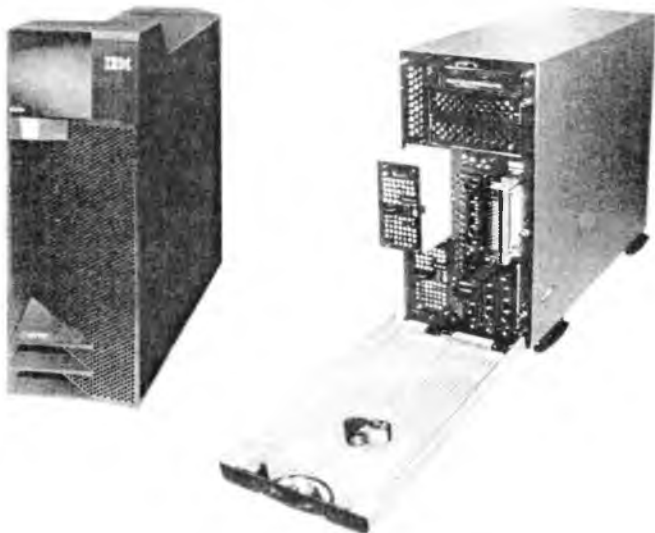


Рис. 2.110. Підлогові (башта) варіанти сервера

Системні плати для серверів будуються на спеціалізованих наборах мікросхем системної логіки (IBM, Sun, ServerWorks GC-SL [HP], Intel E7210 (875P), E7501, Intel Canterwood ES, AMD Opteron 8000), які підтримують багатопроцесорні конфігурації, мають розширені варіанти шини PCI (64-бітне розширення PCI X), Raid-контролери і т.ін.

На рис. 2.111 подано зображення двохпроцесорної системної плати фірми Asus.

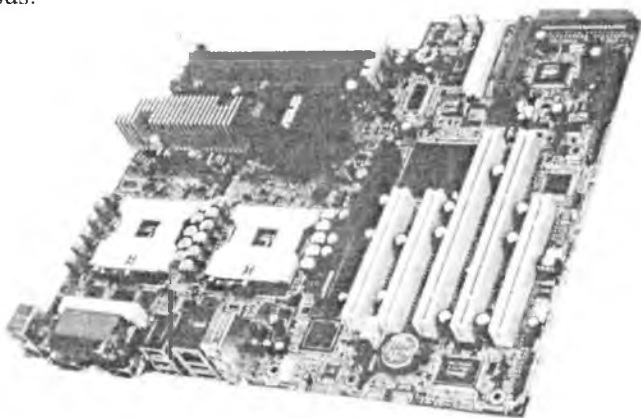


Рис. 2.111. Системна плата Asus PP DLW

Дискова підсистема серверів будується на основі таких інтерфейсів:

- ATA;
- SCSI;
- Serial ATA;
- Serial Attached SCSI.

Оперативна пам'ять серверів (від одиниць гігабайт до терабайт) будується з використанням модулів із кодом корекції помилок (Error Corection Code). Процесор автоматично виправляє одnobітні помилки і видає повідомлення про двобітні. Фірмою IBM розроблена технологія Chipkill, яка автоматично виправляє помилки у кількох бітах і вилучає модулі пам'яті, в яких кількість помилок перевищує критичний рівень.

Окрім мікропроцесорів сімейства x86 для побудови серверів широко використовують RISC-процесори (IBM Power5 [IBM], UltraSPARC [Sun]). Більш висока швидкодія серверів на RISC-процесорах зумовлена 64-бітною архітектурою та скороченою системою команд (скорочується цикл дешифрації команди за рахунок перегляду коротшої таблиці команд). Ще однією особливістю RISC-систем (IBM, HP) є технологія логічних розділів – ресурси сервера одночасно можуть використовувати кілька операційних систем.

### Контрольні запитання

1. Які завдання виконують сервери?
2. Які конструктиви використовують при побудові серверів?
3. Які процесори використовують при побудові серверів?
4. Які складові серверів можна замінювати без вимкнення живлення?
5. Як будується підсистема пам'яті серверів?
6. Які методи використовують при побудові дискової підсистеми серверів?

## 2.7. Живлення та електробезпека ПЕОМ

Для живлення складових системного блоку (і USB-пристроїв) використовують блок живлення, який здійснює перетворення змінної напруги 220В (110В) у постійні напруги +5В (10-50А), +12В (3.5-15А), -12В (0.3-1А), -5В (0.3-0.5А). Напругу +5В використовують для живлення

цифрових мікросхем, +12В – для живлення двигунів дискових пристроїв та для живлення інтерфейсних схем, -12В – для живлення інтерфейсних схем, напруга -5В виробляється для підтримки стандарту ISA і у більшості ПЕОМ не використовується. З метою зменшення розмірів блоки живлення ПЕОМ будуються на основі схем прямого перетворення змінної напруги 220В у постійну напругу близько 400В, яке використовують для живлення імпульсного генератора (робоча частота генератора складає десятки кілогерц). Навантаженням генератора є трансформатор з феритовим осердям. Вторинні обмотки трансформатора з'єднані з випростувачами, які і виробляють напруги +5В, -5В, +12В, -12В. У більшості випадків блоки живлення обладнані перемикачем робочої напруги 220В-110В, однак деякі моделі можуть житися лише від напруги 110В. У цьому випадку доводиться користуватись перетворювачами 220В-110В (трансформаторами або тиристорними конверторами). Потужність, споживана блоком живлення від електромережі, залежить від моделі і змінюється від 150Вт до 500Вт. Цю обставину слід враховувати при додатковому встановленні або заміні дисків (гнучких, жорстких, оптичних). Для охолодження елементів блоку використовують вентилятор, який живиться від напруги +12В. Для усунення завад іншим електроспоживачам, які може створити імпульсний блок живлення ПЕОМ, на вході блоку живлення встановлений ємнісний фільтр.

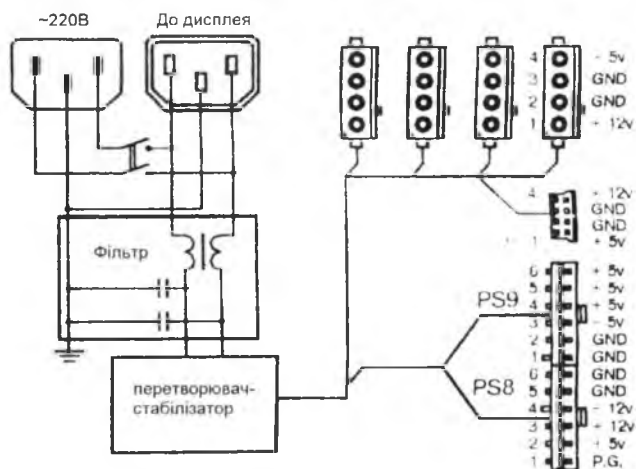


Рис. 2.112. Блок-схема блока живлення стандарту АТ

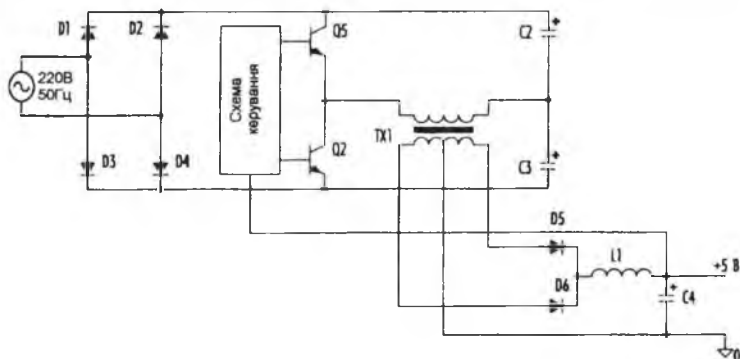


Рис. 2.113. Спрощена схема ланки “перетворювач-стабілізатор” каналу 5В

Вимкнення/ввімкнення блока живлення стандарту АТ здійснюється механічним перемикачем.

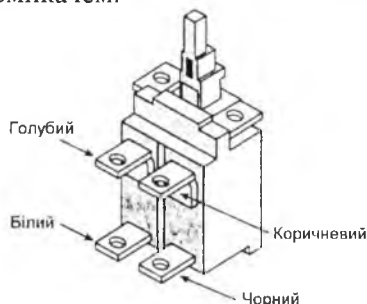


Рис. 2.114. Механічний вимикач живлення

Крім стандарту АТ в IBM-подібних ПЕОМ використовують блоки живлення стандарту АТХ. Блоки живлення стандарту АТХ мають електронний вимикач і замість двох роз'єдтів PS8, PS9 для подання напруг на системну плату використовують двохрандне роз'єдття.

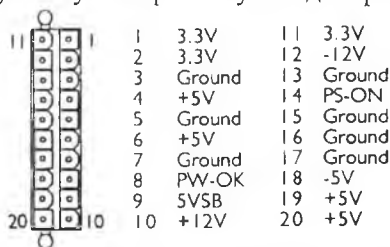


Рис. 2.115. Двохрандне роз'єдття для подання напруг на системну плату стандарту АТХ

Якщо ПЕОМ увімкнена в електромережу без заземлюючого дроту (двополюсна вилка), то конденсатори фільтра блока живлення створюють на корпусі напругу  $\sim 110\text{В}$ , яка може складати небезпеку для людини (і інтерфейсних схем зовнішніх пристроїв вводу/виводу).

На рис. 2.116 зображено правильне приєднання ПЕОМ і периферійного пристрою до електромережі з використанням триполюсних вилок.

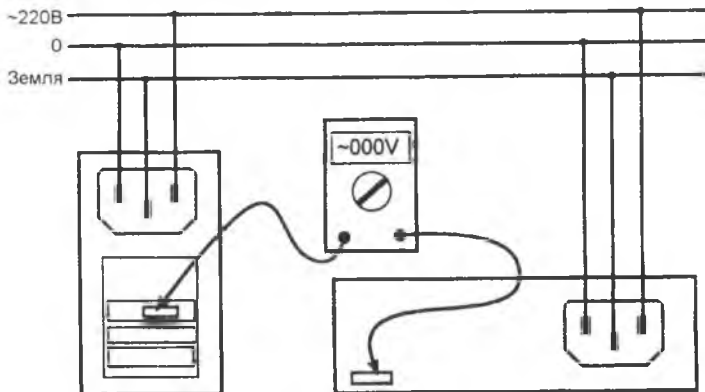


Рис. 2.116. Приєднання ПЕОМ і периферійних пристроїв до електромережі

Якщо у вас не встановлена достатня кількість стаціонарних євро-розеток, доцільно придбати мережевий фільтр (рис. 2.117, а) з певною кількістю розеток (від 3 до 5), які, окрім виконання функції подовжувача, спроможні гасити стрибки напруги до 95% (рис. 2.117, б).



Рис. 2.117. Мережевий фільтр з вимикачем та п'ятьма євро-розетками:  
а – зовнішній вигляд; б – вигляд фільтра зі знятою кришкою

Для забезпечення безперервного живлення ПЕОМ використовують блоки безперервного живлення (Back-Ups), які деякий час після аварійного вимкнення змінної напруги 220В спроможні забезпечувати живлення ПЕОМ змінною напругою за рахунок вбудованих акумуляторів



та електронних схем перетворення постійної напруги 12В у змінну 220В. Розрізняють два види блоків безперебійного живлення:

- комутовані (рис. 2.118);
- прямого перетворення (рис. 2.119).

У комутованих блоках безперебійного живлення перетворювач 12В -> 220В вмикається лише при пропаданні змінної напруги 220В. У блоках прямого перетворення перетворювач 12В -> 220В працює постійно, а напруга з мережі змінного струму використовується лише для підзарядження акумуляторів. Основною характеристикою блоків безперебійного живлення є тривалість забезпечення живлення (від 6 хвилин до трьох годин) на номінальному навантаженні (165Вт-3650Вт).

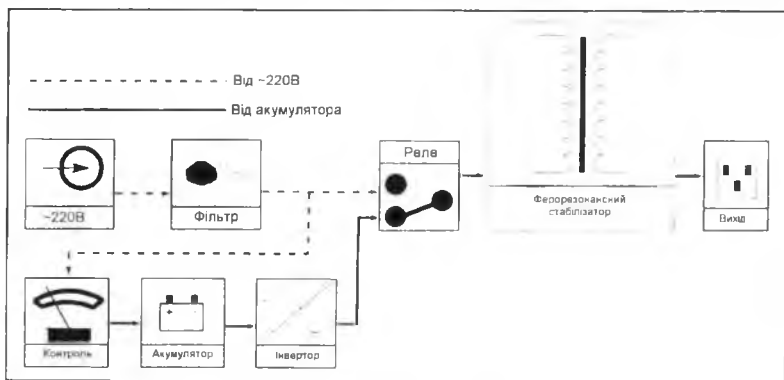


Рис. 2.118. Блок-схема комутованого блока безперебійного живлення

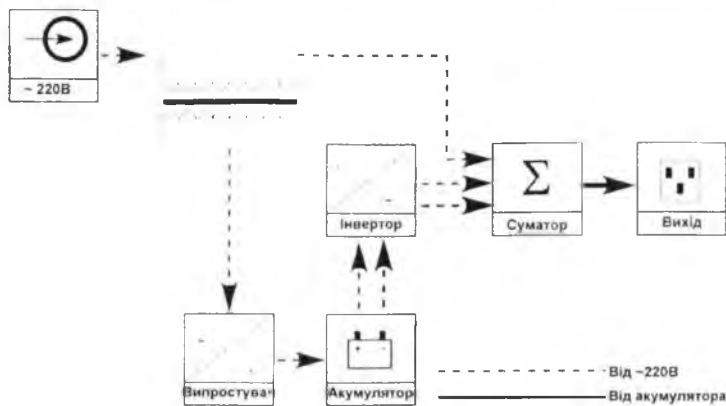


Рис. 2.119. Блок безперебійного живлення прямого перетворення

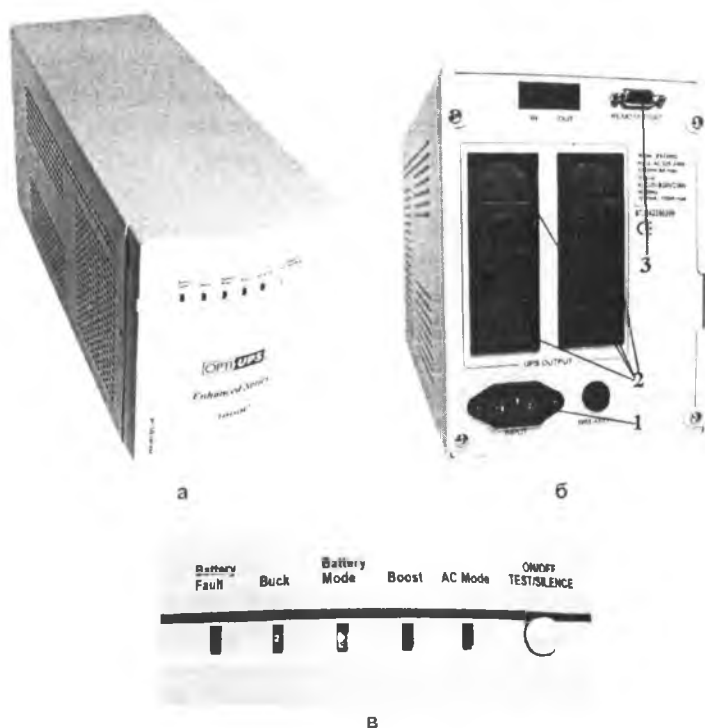


Рис. 2.120. Зовнішній вигляд блока безперебійного живлення (1 – вхід  $\sim 220$  В, 2 – вихід  $\sim 220$  В, 3 – роз'єм для кабелю ініціації вимкнення ПЕОМ): а – вигляд спереду; б – вигляд ззаду; в – індикатори блоку

Деякі блоки безперебійного живлення при зникненні змінної напруги 220В крім подачі звукового сигналу можуть активізувати програму вимкнення ПЕОМ (через спеціальний кабель, під'єднаний до системного блоку).

Для забезпечення енергопостачання переносних ЕОМ переважно використовують акумулятори (у приміщеннях можна використовувати зовнішні блоки живлення), які час від часу потребують підзарядження. Останнім часом активно ведуться роботи по впровадженню альтернативних джерел живлення:

- паливні елементи (переносні, планшетні і кишенькові ПЕОМ) – енергія окислення метанолу в паливному елементі перетворюється на електричну;

- сонячні батареї використовують для підзарядження акумуляторів кишенькових ПК і комунікаторів;
- перетворювачі зусиль м'язів у електричну напругу використовують для підзарядження акумуляторів переносних ЕОМ.



### Контрольні запитання

1. Які напруги виробляє блок живлення ПЕОМ?
2. Що є причиною виникнення міжкорпусної напруги у випадку використання електророзеток без контактів занулення?
3. Як вимикається ПЕОМ стандарту AT?
4. Як вимикається ПЕОМ стандарту ATX?
5. Поясніть принципи роботи безперебійних блоків живлення.
6. Які пристрої використовують для живлення переносних ПЕОМ?

## РОЗДІЛ III.

### АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ ПЕОМ

#### 3.1. Відеосистема

Персональні комп'ютери здобули широке визнання завдяки інтерактивності спілкування з користувачем. Інтерактивність передбачає наявність пристроїв оперативного введення інформації (клавіатура, миша і т. ін.) та виведення інформації. Основним пристроєм виводу інформації (текстової і графічної) є дисплей. Дисплей під'єднується до системної шини ПЕОМ через графічний контролер. Відеосистема IBM PC розрахована на растровий метод формування зображень. Растр складається з певної кількості рядків, кожен рядок складається з великої кількості крапок. Крапки можуть мати різний колір і інтенсивність свічення. Кількість рядків, крапок у рядку, відтінків кольорів, розмір окремої крапки зображення (розмір зерна) та загальні розміри видимої частини зображення (діагональ екрану) визначають споживацькі характеристики дисплея.

##### 3.1.1. Дисплеї і відеопроєктори

За час існування ПЕОМ використовувалися різні моделі дисплеїв, які відрізнялись як зображувальними можливостями (CGA, EGA, HERCULES, VGA, SVGA), так і фізичними принципами формування зображення (свічення люмінофора у електроннопроменевій трубці, газоплазмові дисплеї, рідкокристалічні дисплеї і т.ін.).

Стандарт	Кількість крапок у рядку	Кількість рядків	Кількість кольорів	Примітка
<b>CGA</b>	320 (640)	200	16 (2)	Застаріло
<b>EGA</b>	640	350	16	Застаріло
<b>Hercules</b>	720	348	2	Застаріло
<b>VGA</b>	640	480	16-4294967296	Застаріло
<b>xVGA</b>	800-2048	600-1536	16-4294967296	

На рис. 3.1 зображені схеми формування растрів з прогресивною (Non Interlaced) та черезрядковою розгорткою (Interlaced).

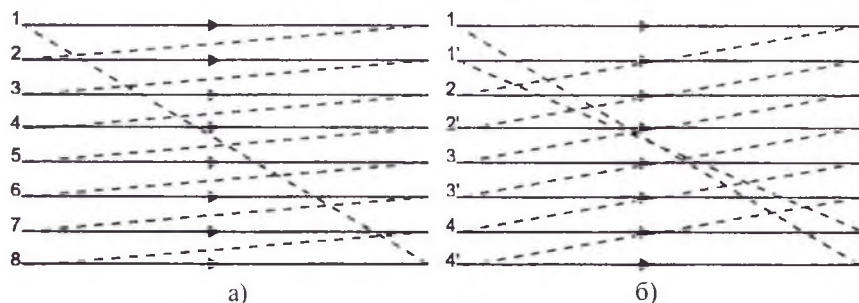


Рис. 3.1. Схема формування рядків зображення на екрані дисплею (а – прогресивна розгортка, б – розгортка через рядок)

Розгортка через рядок дозволяє вдвічі зменшити кадрову частоту (здешевлює графічний контролер), однак тонкі горизонтальні лінії при такому методі розгортки відображаються з “мерехтінням” зображення, а очі користувача швидко втомлюються.

Кольорові зображення в електронно-променевих трубках формуються змішуванням трьох основних кольорів – червоного, зеленого, синього. Люмінофори (речовини, що світяться під дією пучка електронів) з такими кольорами свічення наносяться на скран дисплея так, як це показано на рис. 3.2-3.3. Більша чіткість (але менша яскравість) зображення досягається у дисплеях з так званою “мозаїчною” маскою. Переднє скло звичайної електронно-променевої трубки є частиною сфери (шлях електронів до різних ділянок екрану дисплею має бути однаковим!). На скранах дисплеїв з великою діагоналлю кривизна переднього скла стає доволі відчутною, що призводить до підвищеного навантаження на очі користувача ПЕОМ. Тому виробники дисплеїв за рахунок ускладнення електроніки дисплею стали використовувати різні типи дисплеїв з покращеною геометрією переднього скла (dinaflat, natural flat, flatron). Окрім дисплеїв на електронно-променевих трубках останнім часом комп’ютерні фірми пропонують плоскі дисплеї (TFT-дисплеї), в яких зображення формується підсвічуванням (червоним, синім, зеленим кольором) маленьких комірок (у перерізі від 0,26 мм до 0,42мм). На рис. 3.4 подано зображення комірки у режимі пропускання і затримки світла.

Останнім часом ведуться роботи по створенню електролюмінісцентних органічних плівок (Organic Light Emitting Device), на рис. 3.5 подано зображення одного елемента зображення (pixel). На їх основі

створюють плівкові дисплеї, які можна згорнути в рулон. Такі дисплеї вже використовують у переносних пристроях.

Для формування великоформатного зображення на екранах фірми пропонують портативні, переносні та стаціонарні проектори. За конструкцією проектори поділяють на просвітні (див. рис. 3.6) і мікродзеркальні (див. рис. 3.7)

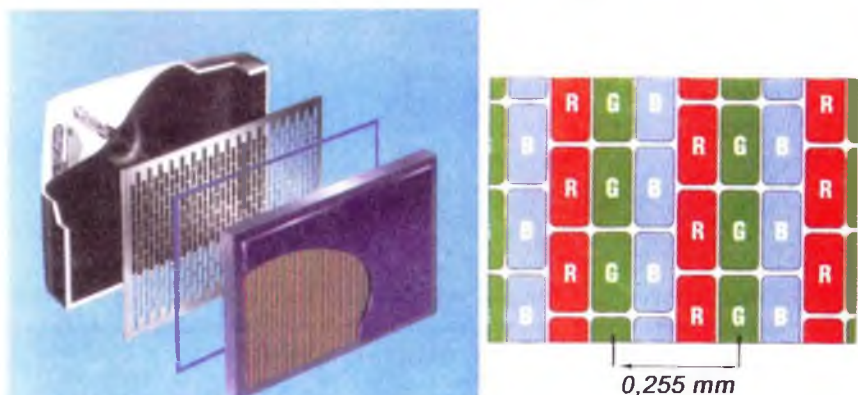


Рис. 3.2. Дисплей із шпаруватою маскою

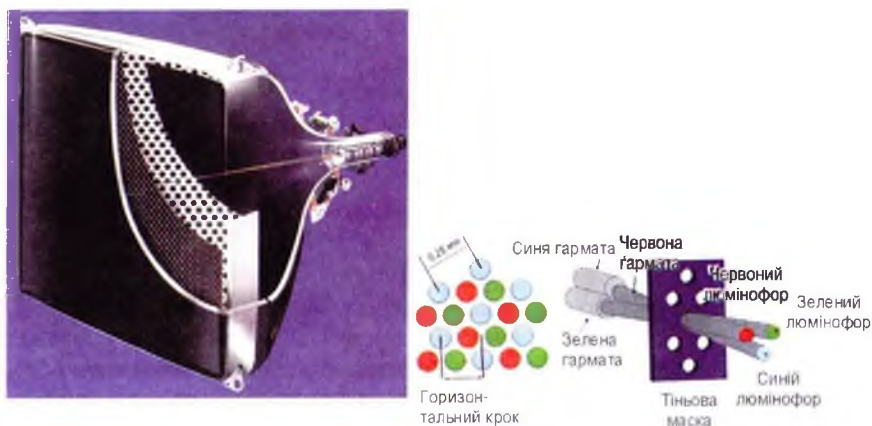


Рис. 3.3. Дисплей з мозаїчною маскою

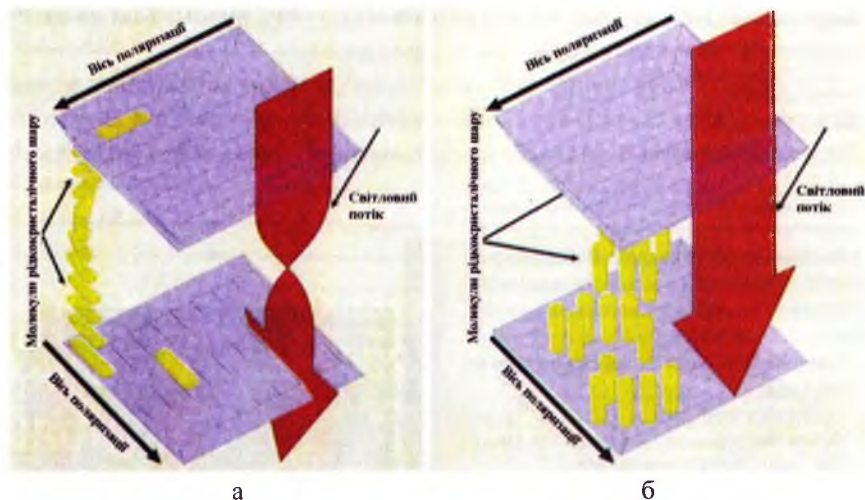


Рис. 3.4. Рідкокристалічна комірка у режимах пропускання і затримки світла:  
а) світло проходить через комірку (висока напруга); б) світло не проходить через комірку (низька напруга)

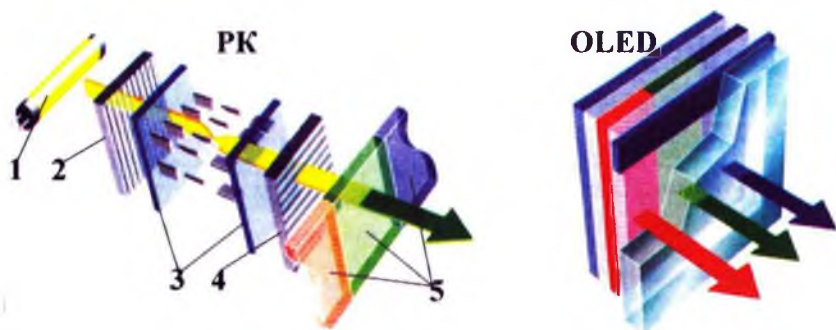


Рис. 3.5. Структура рідкокристалічного дисплея і дисплея на електролюмінесцентній органічній плівці:

1 – лампа білого світла, 2 – поляризатор (вирізає із світлового потоку) хвилі із певною площиною поляризації, 3 – рідкокристалічна комірка (повертає площину поляризації світлового потоку на кут повороту пропорційний напрузі, яка прикладається до комірки), 4 – аналізатор (вихідний світловий потік, пропорційний косинусу кута між площиною поляризації світлового потоку і напрямком орієнтації аналізатора), 5 – світлофільтри



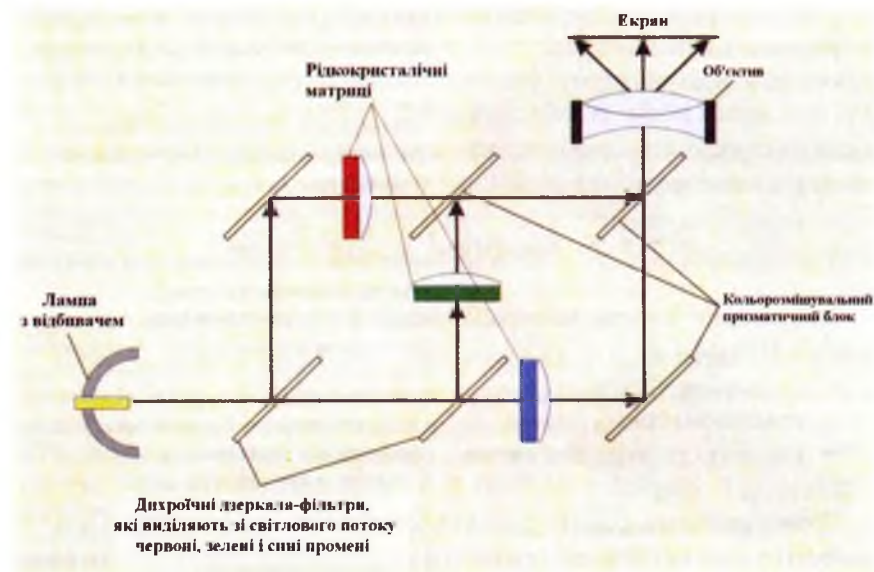


Рис. 3.6. Блок-схема проєктивного проектора

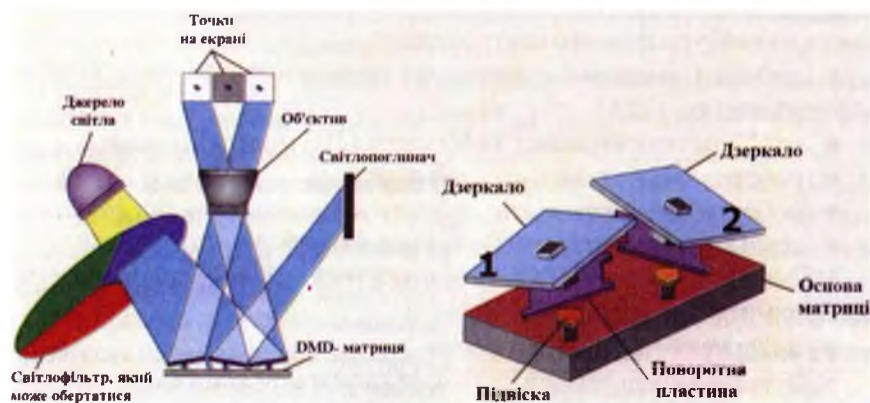


Рис. 3.7. Блок-схема мікродзеркального проектора:

1 – дзеркало відбиває світло в об'єктив;

2 – дзеркало відбиває світло на світлопоглинач

Дисплеї на електронно-променевих трубках займають багато місця на столі і є джерелом шкідливих впливів на людину (рентгенівське



випромінювання, електромагнітне й електростатичне поле, мерехтіння зображення). При виконанні правил безпечної роботи шкідливі впливи можна звести до мінімуму (але не до нуля!). Плоскі дисплеї не створюють шкідливих полів, їх зображення не мерехтить, вони займають мало місця на столі, однак ціни на плоскі дисплеї у кілька разів вищі від цін на дисплеї з електронно-променевими трубками.

### 3.1.2. Графічні контролери

Зображення, видиме на екрані дисплея, формується кількома складовими ПЕОМ:

- дисплеєм;
- графічним контролером;
- шиною, до якої під'єднано графічний контролер (AGP, PCI, PCIExpress [у старіших моделях ISA, VLB, MCA]).

Кожна складова робить свій внесок у якість зображення. При наявності на системній платі порта AGP доцільніше встановлювати плату графічного контролера для AGP, ніж для шини PCI. Вибір конкретної плати слід узгоджувати з планами використання ПЕОМ. Спробуємо перерахувати основні напрями використання ПЕОМ, які істотно впливають на вибір графічного контролера:

- робота з текстом і статичними графічними образами (офісне застосування);
- графічне проектування з об'ємним (3D) моделюванням;
- робота з цифровими фотографіями;
- робота з цифровим відео;
- комп'ютерні тренажери і комп'ютерні ігри.

Масовий користувач ПЕОМ у більшості випадків має справу з дуже поширеними застосуваннями:

- використання ПЕОМ в офісі;
- перегляд відеофільмів на ПЕОМ (домашній кінотеатр);
- комп'ютерні ігри.

Для офісних застосувань важливими параметрами відеосистеми є:

- роздільна здатність;
- глибина (кількість відтінків) кольору;
- чіткість зображення;
- частота перемальовування зображення (кадрова частота).

Прийнятними для роботи з текстом і статичними графічними образами показниками є:

- роздільна здатність 1024×768;
- кадрова частота на менше 85 Гц.

Чіткість зображення при високих значеннях роздільної здатності (1280×1024 і вище) і кадрової частоти (більше 85 Гц) залежать від якості електронних кіл графічного контролера, тому доцільно вибирати графічний контролер відомого виробника. Доволі часто виробники системних плат розміщують графічний контролер на системній платі (інтегрований графічний контролер).

Використання ПЕОМ для комп'ютерних ігор та перегляду відеофільмів вимагає набагато потужніших графічних контролерів, ніж офісні застосування. Тут важливими параметрами є:

- частота графічного процесора контролера (250 МГц, 270 МГц, 300 МГц, 500 МГц);
- обсяг відеопам'яті (64 МБ, 128 МБ, 256 МБ) і її частота (444 МГц, 500 МГц, 550 МГц, 650 МГц);
- апаратна підтримка функцій T&L (Transformation and Lighting – трансформації і розрахунку освітленості) і декодування MPEG2/DVD.

З появою інтерфейсу PCIExpress (див. 2.1.5) графічна підсистема ПЕОМ зазнала деяких новацій:

- можливість встановлення двох графічних контролерів для паралельної роботи на кадром – технологія SLI (Scalable Link Interface, фірма Nvidia);
- використання прямого доступу до оперативної пам'яті (DMA) через швидкий інтерфейс PCIExpress з метою використання частини системної оперативної пам'яті як відеопам'яті – технологія TurboCache фірми Albatron Technology.

Розглянемо ці нововведення докладніше. За технологією TurboCache на бюджетному графічному контролері розташовано 16 Мбайт відеопам'яті, яких достатньо для роботи з офісними документами. При роботі з 3D графікою 128 Мбайт відеопам'яті виділяються у системній пам'яті (якої має бути не менше 512 Мбайт). При використанні системної логіки Intel 915 швидкість обміну графічного контролера з відеопам'яттю через DMA складатиме 4 Гб/с в обох напрямках. У 2005 р. фірма Albatron Technology планує випустити три моделі – TC6200, TC6200U і TC6200Q.

Технологія SLI від NVIDIA може працювати двома способами:

- перший – PGC (Parallel Graphics Configuration), який полягає у поділі кадра на дві горизонтальні частини з динамічним визначенням розміру. Верхню частину опрацьовує перший контролер (master, або primary),

до нього під'єднується дисплей. Нижню частину кадра опрацьовує другий контролер (slave, secondary). Спочатку кожний контролер опрацьовує 50% площі кадра, але якщо верхня частина екрану створюється контролером швидше, ніж нижня, то площа верхньої частини збільшується для створення правильного балансу навантаження на обидва контролери одночасно (або навпаки);

- другий – AFR (Alternate Frame Rendering), який полягає у створенні цілого кадру кожним контролером по черзі з наступним суміщенням послідовності кадрів у буфері кадрів. Для синхронізації роботи контролерів вони сполучаються з'єднувальною платою.



Рис. 3.8. З'єднувальна плата для технології SLI

Слід зауважити, що для реалізації технології SLI потрібно мати системну плату з двома гніздами PCIExpress x16 і потужний блок живлення (не менше 400 Вт).

На відміну від Nvidia, фірма Asus розробила графічний контролер з двома графічними процесорами.



Рис. 3.9. Графічний контролер з двома процесорами

На рис. 3.10 подана схема з'єднання графічного контролера з дисплеєм або відеопроєктором.

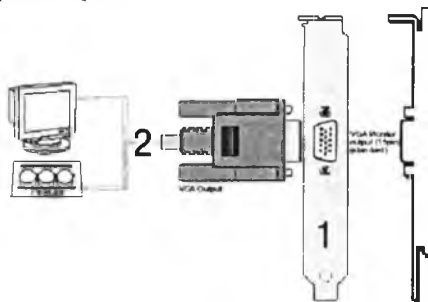


Рис. 3.10. Приєднання дисплея або відеопроєктора до графічного контролера

Більш дорогі графічні контролери дозволяють під'єднувати, крім дисплея або відеопроєктора, додаткове обладнання (телевізор, відеоманітофон, відеокамеру, TV-приймач, DVD-програвач та ін.). На рис. 3.11 подана схема з'єднань графічного контролера з TV входами і виходами.

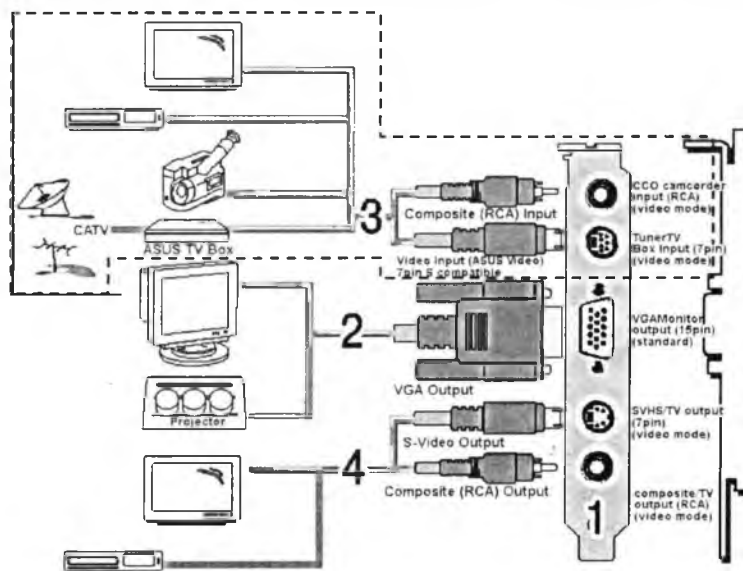


Рис. 3.11. Схема приєднання відеообладнання до графічного контролера:

- 1 – планка плати розширення, 2 – кабель для приєднання дисплея або відеопроєктора, 3 – кабель для введення телевізійного сигналу, 4 – кабель для виведення телевізійного сигналу

Деякі виробники графічних контролерів комплектують їх стереоокулярами для формування стереозображень. На рис. 3.12 подана схема з'єднань графічного контролера, стереоокулярів і дисплея.

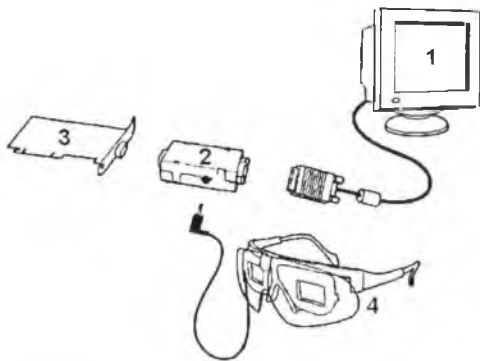


Рис. 3.12. Схема з'єднань графічного контролера, стереоокулярів і дисплея: 1 – дисплей, 2 – синхронізатор, 3 – графічний контролер, 4 – стереоокуляри

Зведення до спільного показника якості характеристик дисплея, відеопам'яті і графічного процесора досить непроста справа навіть для фахівців. Для об'єктивного оцінювання відсисистеми доцільно використовувати тестові програми:

- Nokia-Test – тест для визначення характеристик дисплея;
- 3D Mark 2000 SE – тест для визначення 3D-продуктивності ПЕОМ.

Програми тестування обладнання ПЕОМ можна знайти на оптичних дисках, які додаються до комп'ютерних журналів (CHIP, Компьютер-Пресс та ін.), та в Інтернеті.

### Контрольні запитання

1. Які типи дисплеїв ви знаєте?
2. Які фактори, шкідливі для людини, проявляються при роботі дисплея на електронно-променевій трубі (кінескопі)?
3. Які типи відеопроєкторів ви знаєте?
4. Які типи системних інтерфейсів використовують для графічних контролерів?
5. Поясніть основні принципи технології SLI (Scalable Link Interface).
6. Поясніть основні принципи технології TurboCache.

## 3.2. Дискава підсистема

Магнітні та оптичні диски використовують для довготривалого зберігання інформації (інформація на дисках зберігається і після вимкнення живлення ПЕОМ!). Якщо інформація, з якою ви працюєте, є дуже важливою, доцільно періодично проводити її архівування. При великих обсягах архіву використовують спеціальні пристрої архівації.

### 3.2.1 Принципи магнітного запису

Запис інформації на магнітні диски являє собою створення на феромагнітному покритті підкладки (гнучкого пластикового диску, алюмінієвого або скляного твердого [hard] диску, пластиковій стрічці) ділянок з різними напрямками намагнічення.

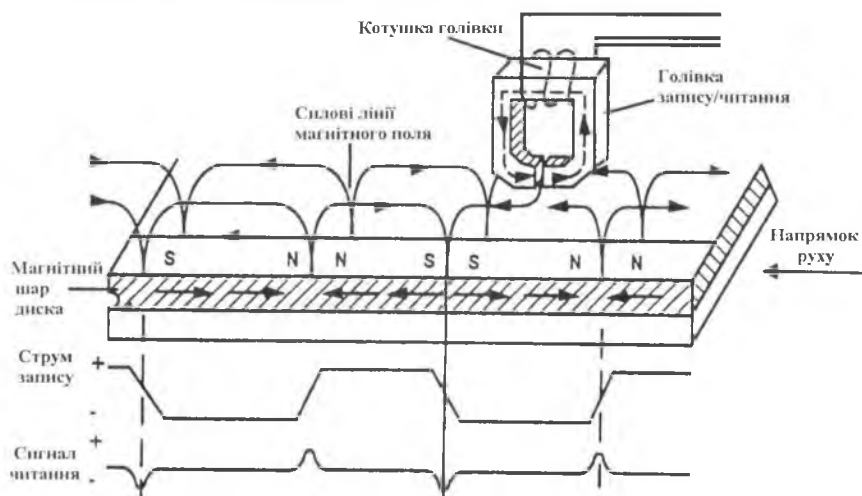


Рис. 3.13. Запис інформації на магнітний носій

Для читання інформації використовують універсальну голівку запису/читання або магніто-резистивну голівку (ЖМД)

При переміщенні намагніченого носія (магнітного диска або стрічки) повз універсальну голівку в ній виникають імпульси (рис. 3.13) у момент проходження біля голівки межі розділу областей з різними напрямками намагніченості (закон електромагнітної індукції). У магніторезистивних голівках використовують залежність опору магніто-

резистивного матеріалу від напруженості магнітного поля. Впровадження магніто-резистивних голівок дозволило збільшити щільність запису (зменшити розмір бітової ділянки) інформації.

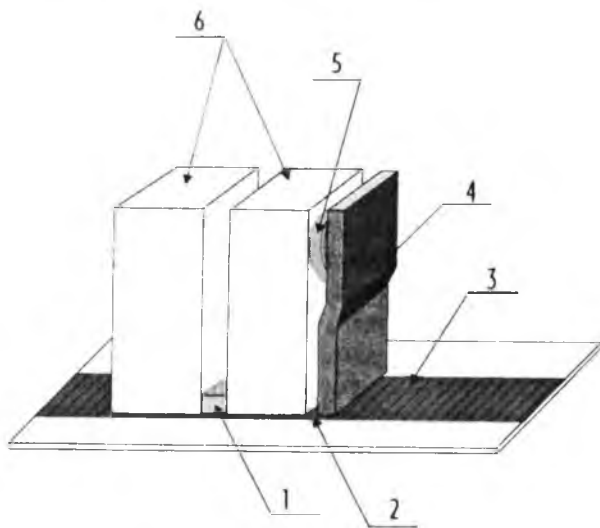


Рис. 3.14. Будова магніто-резистивної голівки:  
1 – магніто-резистивний матеріал, 2 – щілина магнітопроводу,  
3 – ферромагнітне покриття, 4 – магнітний екран, 5 – котушка,  
6 – магнітопровід

Ще більшої щільності запису інформації можна досягти завдяки переходу на перпендикулярний запис інформації – коли вектор намагніченості бітової ділянки розташований перпендикулярно до робочої поверхні.

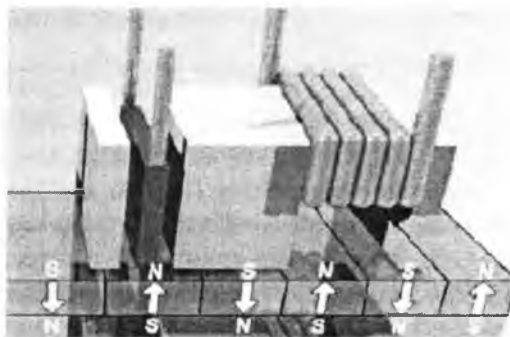


Рис. 3.15. Перпендикулярний метод запису інформації

Завдяки перпендикулярному методу запису місткість ЖМД у найближчі роки планують збільшити до одиниць терабайт.

### Магнітні диски

Магнітні диски поділяють на гнучкі (ГМД, дискети) та жорсткі (ЖМД [“вінчестери”]). Гнучкі магнітні диски для IBM-подібних ПЕОМ випускались двох діаметрів – 5,25” (застаріло) і 3,5”. У таблиці подані всі формати ГМД для IBM-подібних ПЕОМ.

Дискета	Місткість	Примітка
5,25” DD	360K	Застаріло
5,25” HD	1,2M	Застаріло
3,5” DD	720K	Застаріло
3,5” HD	1,44M	
3,5” ED	2,88M	

Інформація на ГМД зберігається на концентричних доріжках (Track), які розділені на сектори.

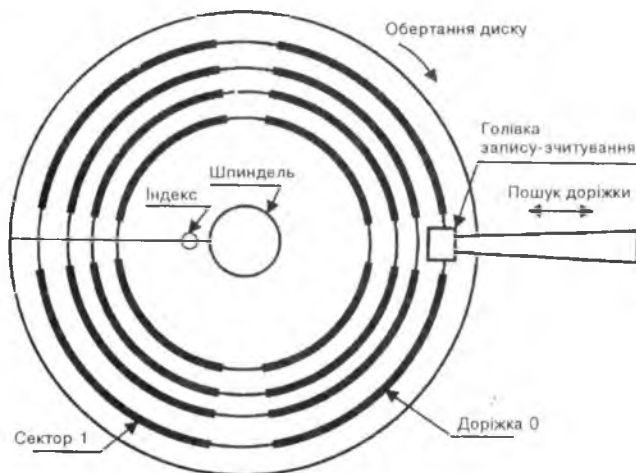


Рис. 3.16. Схема розміщення інформації на магнітному диску

Під час читання інформації з ГМД голівка читання/запису інформації торкається поверхні гнучкого пластикового диска (диск обертається зі швидкістю 360 обертів за хвилину). Для захисту ГМД від забруднень їх тримають у пластикових коробках (конвертах).



ГМД потрібно захищати від зовнішніх магнітних полів, які можуть поспути інформацію на дискеті. Принципи роботи жорстких магнітних дисків подібні до принципів роботи гнучких магнітних дисків, однак феромагнітне покриття наносять на металеві або скляні (жорсткі) пластини, які розміщують у герметизованій коробці з добре очищеним від пилу повітрям. Пакети магнітних дисків (прообраз сучасних ЖМД) стали широко використовувати на великих ЕОМ з середини 60-х рр. XX ст. На ЕОМ серії ЕС-10хх (радянський аналог IBM-360/370) використовували пакети алюмінієвих дисків місткістю 7Мб (5 пластин), 29Мб (10 пластин), 100Мб (10 пластин), 200Мб (10 пластин). Діаметр однієї пластини складав 355мм. Пакети магнітних дисків були змінними, вага 100 мегабайтного пакету була такою, що для його монтажу використовували спеціальний механічний пристрій. Після появи персональних ЕОМ виникла потреба у малогабаритних пристроях запису/читання інформації на магнітні диски. Першим жорстким магнітним диском (ЖМД) діаметру 5,25" став пристрій ST-506 фірми Seagate місткістю 5Мб. Подальший розвиток технології ЖМД полягав у зменшенні розмірів пристрою, збільшенні місткості ЖМД – сотні гігабайт на початку III тисячоліття, збільшенні швидкості обертання пакету дисків – 3600 об/хв, 5400 об/хв, 7200 об/хв, 10000 об/хв, 15000 об/хв, збільшенні швидкості роботи контролера ЖМД.

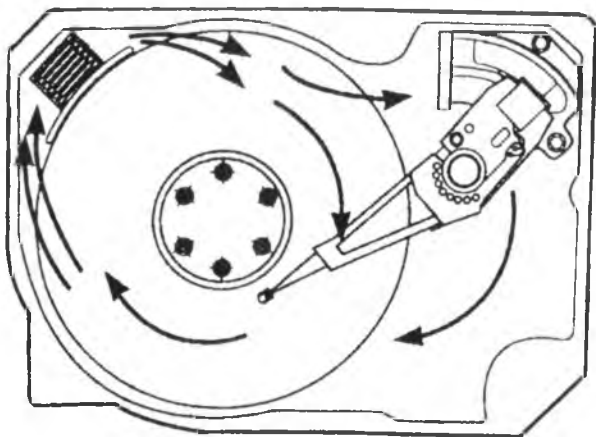


Рис. 3.17. Внутрішня будова ЖМД  
(стрілками вказано напрямки руху повітря при обертанні пакету дисків)

На рис.3.17 подано зображення внутрішньої будови ЖМД, на рис. 3.18 – зображення блоку голівок читання/запису інформації. Під час роботи ЖМД голівки читання/запису рухаються над пластинами на невеликій відстані від пластини завдяки повітряному потоку, що зумовлений обертанням пластин. Для стабілізації відстані між голівкою і пластиною голівку прикріплюють до повзунка з боріздками спеціальної форми (рис. 3.19).

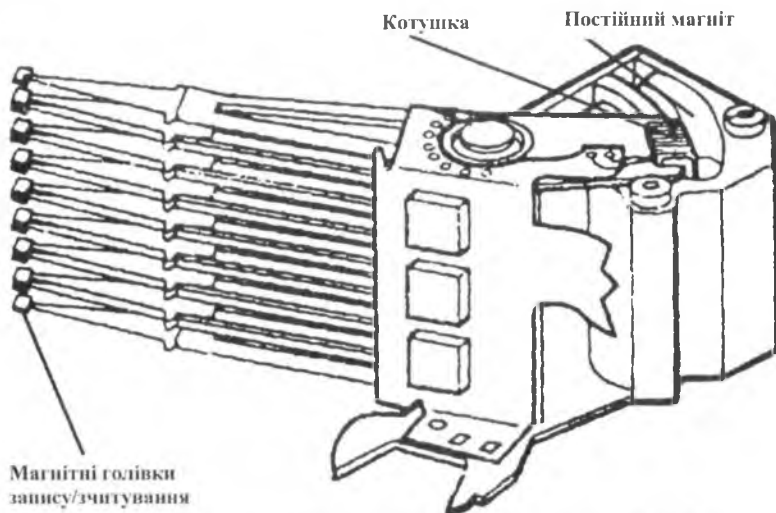


Рис. 3.18. Блок голівок запису/читання інформації ЖМД

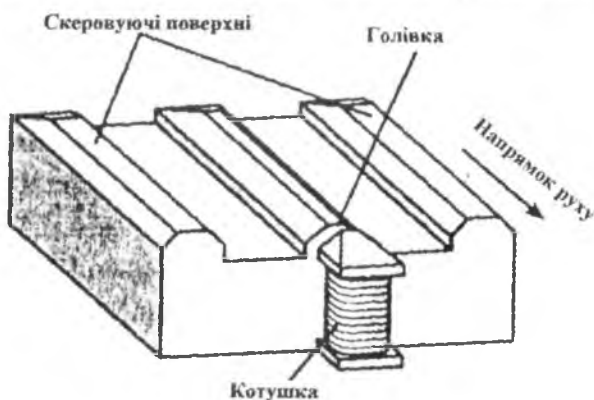


Рис. 3.19. Повзунок з магнітною голівкою



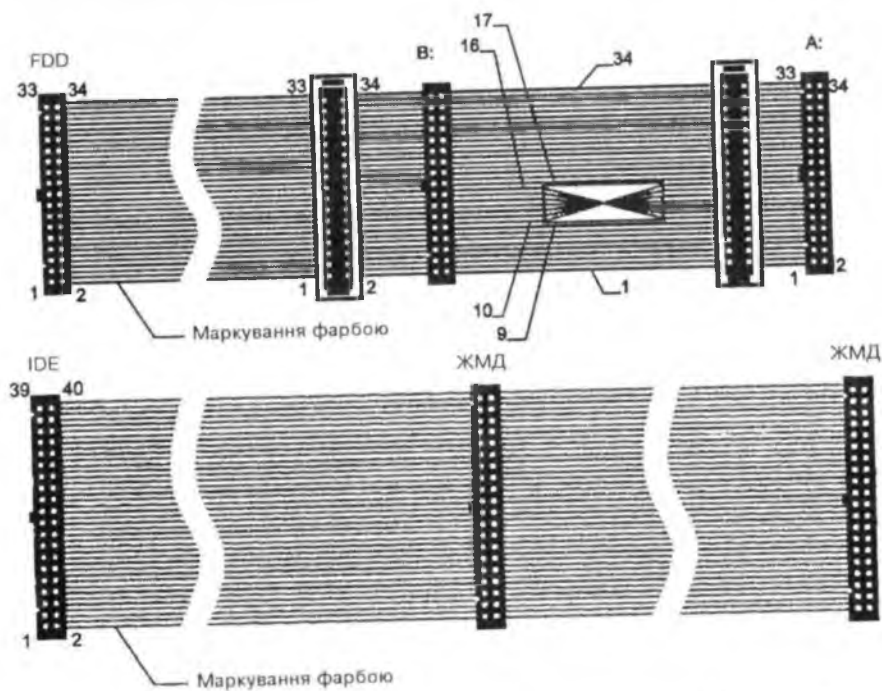


Рис. 3.21. Кабелі для приєднання гнучких та жорстких магнітних дисків

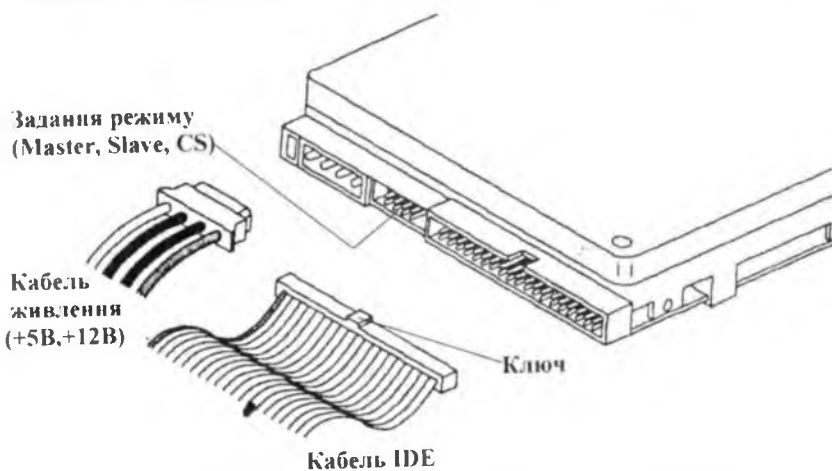


Рис. 3.22. З'єднання ЖМД із силовим і сигнальним кабелями

Якщо на одному кабелі інтерфейсу ATA IDE встановлені два пристрої з різною (високою та низькою) швидкістю обміну інформацією, то це може бути причиною повільної роботи ПЕОМ. У зв'язку з цим пристрій для роботи з оптичними дисками доцільно під'єднувати окремим кабелем до контролера IDE2.

Швидкість обміну інформацією між процесором та ЖМД істотно впливає на швидкодію ПЕОМ. Залежно від набору мікросхем системної логіки ця швидкість може бути різною:

ATA 33 – 33 Мб/с;

ATA 66 – 66 Мб/с;

ATA 100 – 100 Мб/с;

ATA 133 – 133 Мб/с.

З 2004 р. розпочалось серійне виробництво системних плат зі стандартом Serial ATA, у якому замість паралельного використовується послідовне передавання інформації. Це дозволяє перейти на більш високі швидкості передавання інформації (Serial ATA I – 150 Мб/с, Serial ATA II – 300 Мб/с, Serial ATA III – 600 Мб/с.), збільшити до 90 см довжину кабеля та спростити приєднання пристроїв (на кожен пристрій окремий кабель).

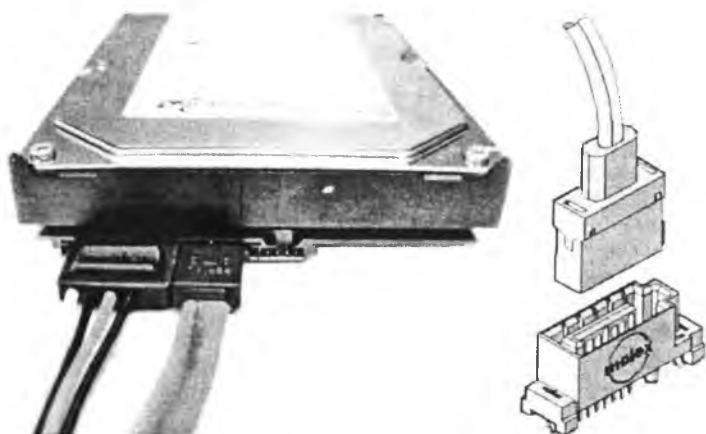


Рис. 3.23. Жорсткий магнітний диск з інтерфейсом Serial ATA і розняття цього інтерфейсу

Для використання ЖМД з паралельними інтерфейсами розроблені перехідники Serial ATA – IDE.

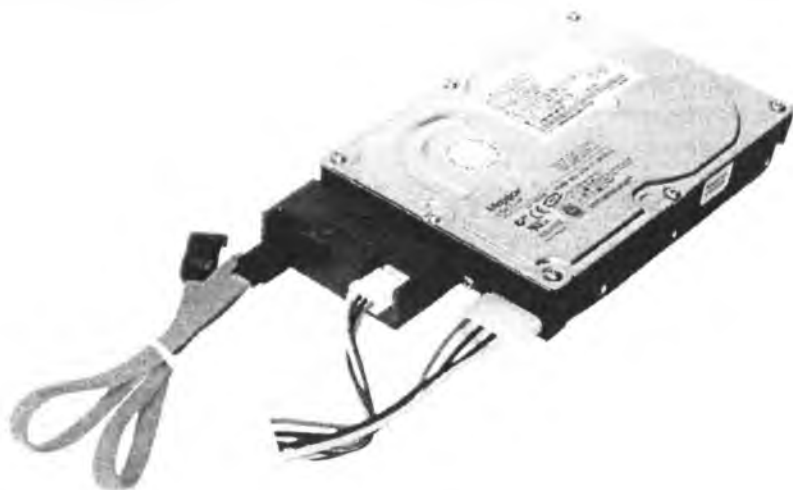


Рис. 3.24. Жорсткий магнітний диск з перехідником для Serial ATA

Якщо потрібно часто переносити інформацію з однієї ПЕОМ на іншу, і ці ПЕОМ не зв'язані мережею, доцільно використовувати контейнер (Mobile Rack) для змінних ЖМД (рис.3.25).



Рис. 3.25. Контейнер (Mobile Rack) для змінних ЖМД

Удосконалення технології магнітного запису привело до появи малогабаритних ЖМД (у поперечному розрізі менше за один дюйм), які використовують у переносних ПЕОМ, цифрових фотоапаратах та ін. На їх основі деякі виробники пропонують зовнішні ЖМД з інтерфейсом USB.

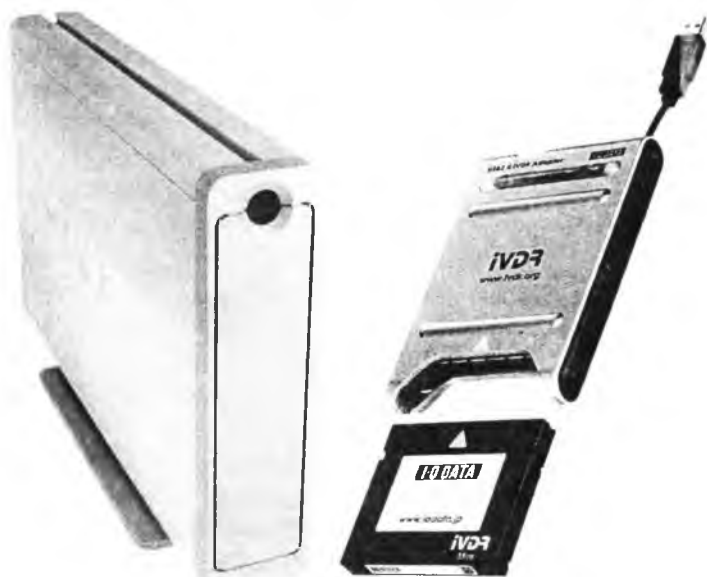


Рис. 3.26. Зовнішні ЖМД з інтерфейсом USB

При зберіганні інформації на файл-серверах потрібно забезпечувати великий обсяг дискової пам'яті і високу швидкість запису/читання інформації. Одним із методів розв'язання цієї проблеми є збільшення місткості ЖМД і підвищення швидкості обертання пакету дисків. Іншим методом є використання RAID-масивів жорстких магнітних дисків (Redundant Array of Inexpensive Disks) – кілька ЖМД із спеціалізованим контролером (як правило, з інтерфейсом SCSI) сприймаються операційною системою як один великий диск. Наявність кількох дисків дозволяє розпаралелити операції запису/читання інформації, що у кінцевому рахунку дозволяє збільшити швидкість запису/читання інформації. Уведення записів контролю парності (зон парності) дозволяють “відчути” і виправити помилки читання з ЖМД. Розглянемо методи розподілу інформації по дисках RAID-масиву.

Приклад організації RAID-масиву нульового рівня із чотирьох ЖМД зображено на рис. 3.27,а. Інформація у RAID-масиві нульового рівня розподіляється по зонах. При отриманні запиту на великий блок інформації, який займає кілька зон, контролер RAID-масиву формує окрему команду читання на кожен диск (але не більше кількості ЖМД у масиві)

і ці команди читання виконуються паралельно. Якщо кількість зон у запиті більша за чотири, то спочатку виконуються команди читання з першої четвірки, потім з другої і т.д. Відсутність зон контролю у RAID-масиві нульового рівня не дає змоги компенсувати помилки читання інформації. При отриманні запитів на кількість інформації, меншу від розміру однієї зони, RAID-масив працює як звичайний ЖМД.

Для підвищення надійності читання інформації з RAID-масиву можна використовувати дублювання інформації. На рис. 3.27,б зображено RAID-масив із восьми ЖМД, причому на ЖМД з п'ятого по восьмий розташовується продубльована інформація. При помилці читання інформації з основної зони використовують інформацію з її копії. Швидкість запису інформації в такому масиві така ж, як у RAID-масиві нульового рівня з чотирма ЖМД. При великих запитах на читання інформації (більше чотирьох зон) контролер RAID-масиву формує команди читання інформації як із основної зони, так і з зони копій (максимально вісім команд).

RAID-масив другого рівня будують із ЖМД, синхронізованих по циліндрах і позиції голівки запису/читання у циліндрі. До 32-бітного слова додають 6 бітів кодів парності (код Геммінга) і кожний біт записують на окремий ЖМД (потрібно мати 38 синхронізованих ЖМД). Загальна швидкодія RAID-масив другого рівня є дуже високою, надійність зберігання інформації також, однак потрібна велика кількість ЖМД і складний, швидкодіючий контролер RAID-масиву (для кожного прочитаного слова потрібно обчислювати контрольну суму!).

RAID-масив третього рівня є спрощенням RAID-масиву другого рівня – кожен біт слова записується на окремий ЖМД. Контролер підраховує біт парності слова і записує його на окремий ЖМД. Як і у RAID-масиві другого рівня, ЖМД синхронізуються по циліндрах і положенні голівки запису/читання у циліндрі. На рис. 3.27,г подана структура RAID-масиву третього рівня з 4-бітним словом.

RAID-масив четвертого рівня є масивом нульового рівня, до якого додали зони парності (рис. 3.27,г). Помилки читання виправляються завдяки інформації, записаній у зоні парності, однак при зміні інформації лише в одній зоні потрібно читати інформацію з решти зон і заново обчислювати контрольну суму для запису в зону парності.

RAID-масив п'ятого рівня зображено на рис. 3.27,д. Зони парності розподіляються по усіх дисках.



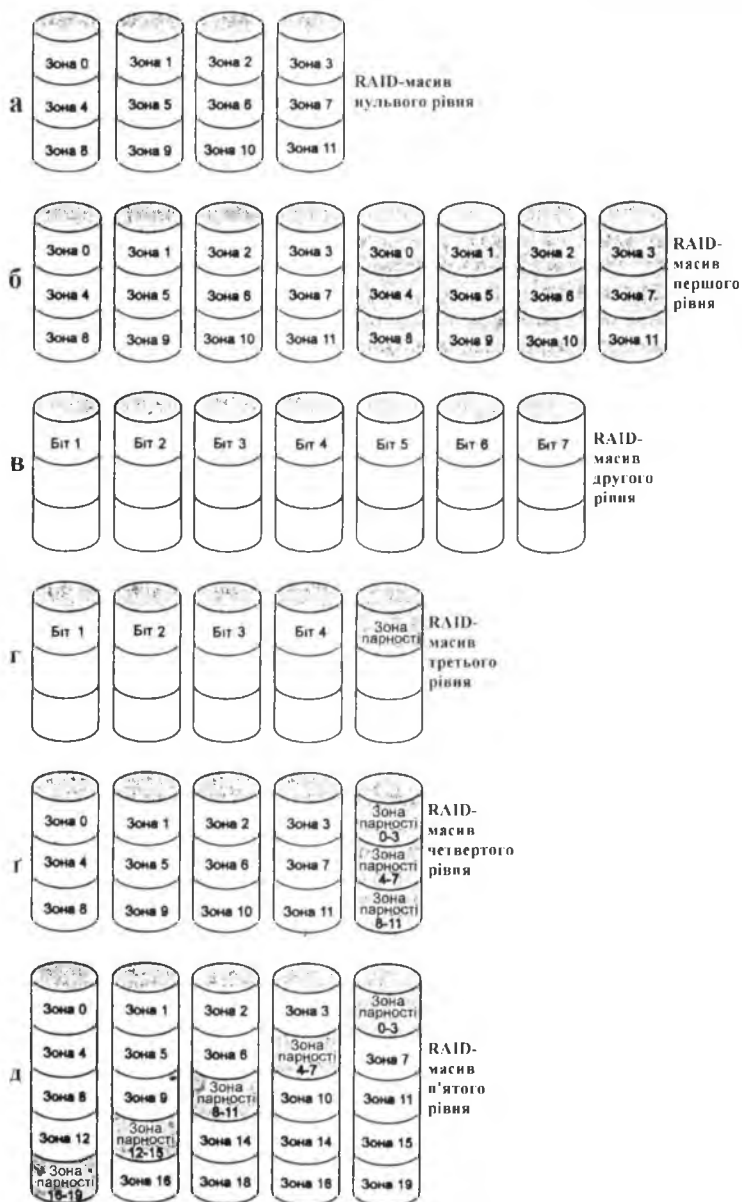


Рис. 3.27. RAID-масиви з нульового по п'ятий рівні

### 3.2.2 Оптичні диски

Зчитувачі інформації з оптичних дисків сьогодні встановлюються у більшості нових ПЕОМ. Найбільш поширеними є так звані CD-ROM (Compact Disk-Read Only Memory) – інформація в них заноситься на диск один раз (штампуванням фольги, яка згодом заливається прозорим пластиком, або записом лазерним променем на спеціально підготовлений чистий диск). На рис. 3.28 показаний фрагмент оптичного диска з записаною інформацією, а на рис. 3.29 – спрощена будова пристрою читання інформації з оптичного диска.

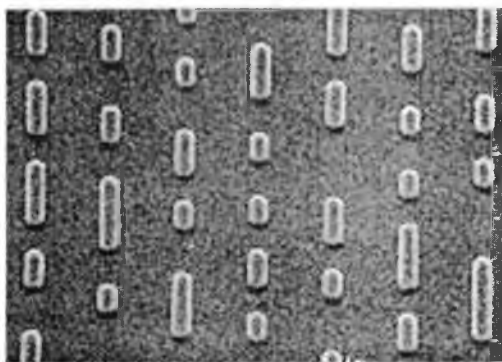


Рис. 3.28. Фрагмент оптичного диска з записаною інформацією

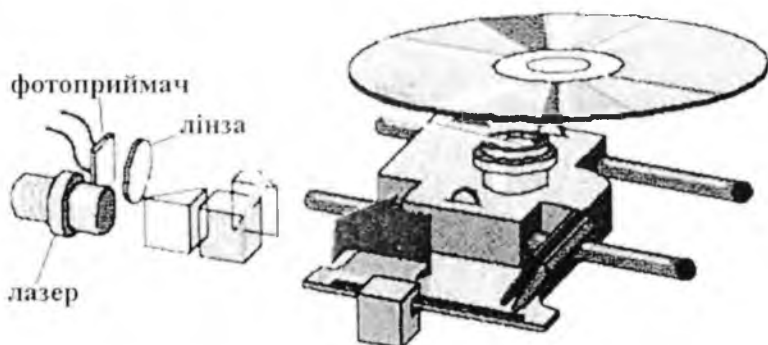


Рис. 3.29. Спрощена будова пристрою читання інформації з оптичного диска

Інформацію на оптичному диску записують по спіральній доріжці. Система слідкування за доріжкою забезпечує фокусування променя лазера на доріжці. Коли у пристрій читання інформації вставляють оптичний

диск, ця система забезпечує виведення променя лазера на початок доріжки у внутрішній частині диска. Читання інформації з диска здійснюється переміщенням променя лазера по спіральній доріжці від внутрішньої частини диска назовні. Для забезпечення постійної швидкості лазерного променя по доріжці кутова швидкість обертання диска є змінною.

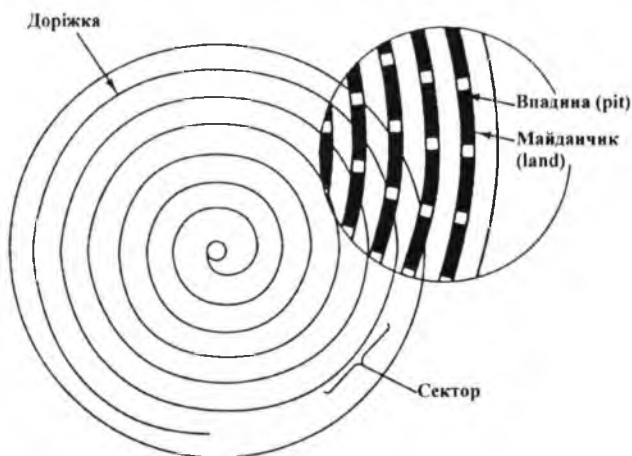


Рис. 3.30. Структура запису інформації на оптичному диску

Промінь від лазера через систему призм скеровується на об'єktiv, який фокусує промінь лазера на фользі. Промінь, відбитий від фольги, через об'єktiv і систему призм скеровується на фотоприймач, який перетворює інтенсивність падаючого на нього оптичного потоку в електричний сигнал. При відбиванні променя від заглиблень на фользі він не потрапляє в об'єktiv.



Рис. 3.31. Відбивання лазерного променя від фольги

Для компенсації помилок, які виникають унаслідок забруднення поверхні оптичного диска, кожен байт кодується 14-бітними символами. Із 42 символів формується фрейм, 98 фреймів формують сектор.

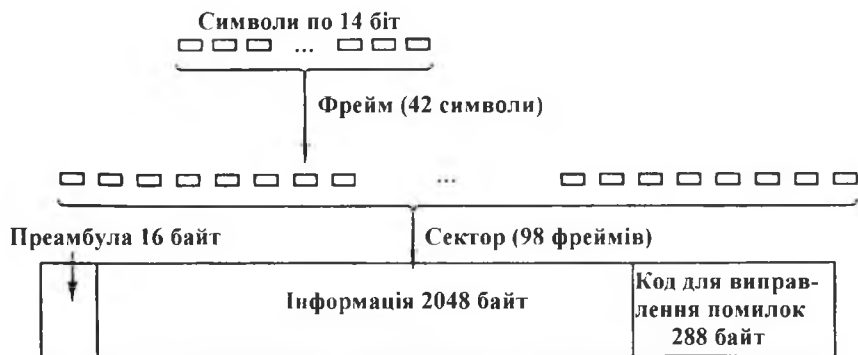


Рис. 3.32. Схема розташування інформації на оптичному диску

У оптичних дисках типу CD-ROM використовують інфрачервоний лазер з довжиною хвилі 780 нм. Довжина бітової ділянки і ширина доріжки повинні бути більшими від довжини хвилі лазера. З метою збільшення місткості оптичних дисків було запропоновано стандарт DVD – для читання інформації використовують два лазери (спочатку застосовували інфрачервоний і червоний, згодом жовтий і зелений). Ще більшу місткість мають диски з голубим і ультрафіолетовим лазерами – Blue Ray диски. Перехід до лазерів з меншою довжиною хвилі дозволив зменшити ширину доріжки і довжину бітової ділянки (рис. 3.33), а отже збільшити щільність запису інформації. Для збільшення місткості DVD-дисків у них розташовують два (односторонній диск) або чотири робочих шари (двосторонній диск). На рис.3.34 зображена взаємодія лазерних променів з робочими шарами.

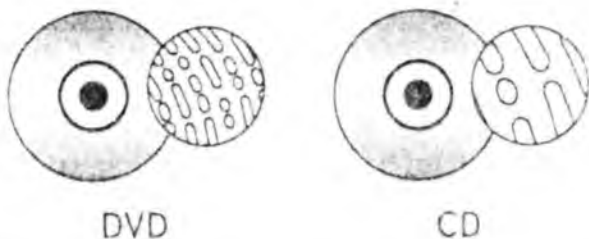


Рис. 3.33. У DVD-дисках розміри штрихів (бітових ділянок) менші, ніж у CD-дисках

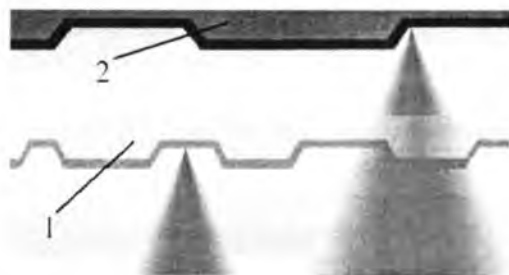


Рис. 3.34. DVD-диск з двома робочими шарами: 1 – відбиваючий шар для довжини хвилі 650 нм (прозорий для 780 нм); 2 – відбиваючий шар для 780 нм

Формат Blu-ray істотно відрізняється від DVD. У ньому використовують нові алгоритми зчитування і обробки інформації, що дозволяє зменшити довжину бітової ділянки ( 0.138, 0.149 або 0.160 мкм) і відстань між доріжками. Реєструючий шар на диску розташований на відстані 0.1 мм от поверхні. Завдяки цьому зменшуються спотворення лазерного променя і час відгуку. Основні характеристики оптичних дисків (CD, DVD, Blu-ray, HD-DVD) подано у наступній таблиці.

Показник	CD	DVD	Blu-ray	HD-DVD
Місткість штампованого (ROM) диска, Гбайт на один шар	0,68	4,7	23,3/25	15
Місткість штампованого (ROM) диска, Гбайт з двома шарами	немає	8,5	46,6/50	30
Місткість диска багаторазового запису/читання (RW) на один шар	0,68	4	23,3/25/27	20
Місткість диска багаторазового запису/читання (RW) з двома шарами, Гбайт	немає	немає	46,6/50/54	32
Місткість диска одноразового запису (R) з одним робочим шаром, Гбайт	0,68	4,7	23,3/25/27	15
Місткість диска одноразового запису (R) з двома робочими шарами, Гбайт	немає	8,5	46,6/50/54	немає
Максимальна місткість існуючих прототипів багат шарових дисків, Гбайт	1,4	8,5	100	45
Довжина хвилі лазера, нм	780	650	405	405

Продовження таблиці

Показник	CD	DVD	Blu-ray	HD-DVD
Апертура	0,45	0,6	0,85	0,65
Потужність променя лазера при читанні, мВт	-	-	0,35	0,5
Захисний шар, мм	1,2	0,6	0,1	0,6
Розмір бітової ділянки, нм	830	410	160/149/138	204
Відстань між доріжками, нм	1600	740	320	400
Швидкість передачі інформації, Мбіт/с	0,15 (1x)	1,385 (1x)	36 (1x)/ 54 (video BD-ROM)	36,5 (1x)
Підтримка Java	немає	немає	є	немає
Пітримувані кодеки	-	MPEG 2	MPEG2/ MPEG4/ AVC/ VC-1	MPEG2/ MPEG4/ AVC/ VC-1
Система захисту інформації	-	CSS	AES	AACS

**Примітки:**

Content Scrambling Sequirity (CSS) – система шифрування вмісту;

Advanced Access Content System (AACS) – поліпшена система доступу до вмісту;

Advanced Encryption Standard (AES) – поліпшена система шифрування;

Audio-Visual Connection (AVC) – засоби опрацювання аудіовізуальної інформації;

Motion Pictures Exprts Group (MPEG) – експертна група з кінематографії;

Virtual Container (VC-1) – стандарт захисту інформації.

Для запису інформації на спеціально підготовлені диски використовують пристрої CD-RW (Compact Disk Read /Write). Розрізняють оптичні диски одноразового запису і багаторазового зчитування (CD-R) та диски багаторазового запису і зчитування (CD-RW). Диск одноразового запису (CD-R) всередині пластикової обкладинки має металевий шар, покритий барвником. Матеріал барвника (ціанін зеленого кольору, або фталоціанін жовтого) є прозорим для променів інфрачервоного лазера, однак при записуванні інформації завдяки більшій потужності записуючого променя барвник втрачає свою прозорість (з'являються плями на доріжці). На рис. 3.35 зображена схема читання інформації з диска CD-R.

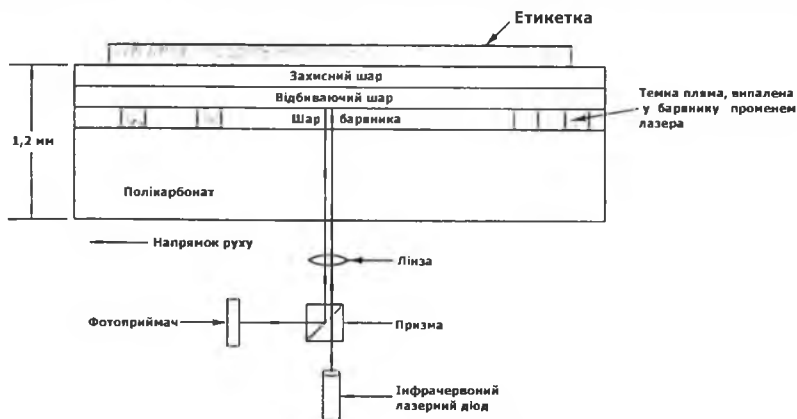


Рис. 3.35. Читання інформації з диска CD-R

Диски багаторазового запису також ґрунтуються на зміні коефіцієнта відбивання робочого шару під дією потужного лазерного променя. Робочий шар дисків CD-RW виготовляють зі сплаву (наприклад Ag-In-Sb-Te), який може перебувати в аморфному і кристалічному станах. Коефіцієнти відбивання аморфного і кристалічного стану дуже відрізняються. При записуванні інформації потужний промінь лазера розігрівляє окремі ділянки робочого шару до температури 500-700 °C, розплавляючи їх. Після охолодження ці ділянки переходять у кристалічний стан. При витиранні інформації увесь диск розігрівается до температури 200 °C, що викликає перехід кристалічних ділянок у аморфний стан. На рис. 3.36 подана внутрішня будова дисків CD-R і CD-RW.

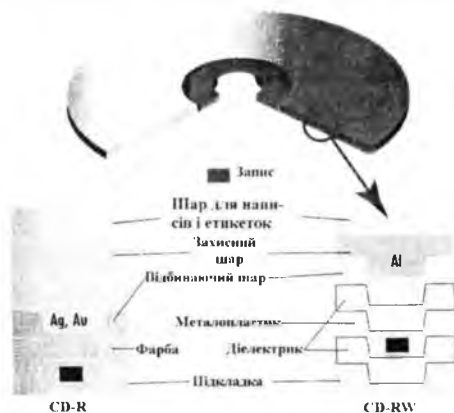


Рис. 3.36. Диск одно- і багаторазового запису інформації

Слід зазначити, що у дисках CD-R і CD-RW робочий шар наноситься у боріздки шириною 0,6 мм. Це потрібно для системи слідування за доріжкою під час запису інформації.

Окрім внутрішніх пристроїв читання інформації з оптичних дисків, які встановлюються в системний блок, використовують зовнішні пристрої. Основним інтерфейсом приєднання внутрішніх пристроїв є інтерфейс IDE, зовнішні пристрої під'єднують до системного блоку за допомогою інтерфейсів USB, FireWare, SCSI.



Рис. 3.37. Пристрої читання інформації з оптичних дисків з відкритим лотком

Важливим показником пристроїв зчитування (і запису) є множник, який вказується у назві пристрою – 52x, 45x тощо. Цей множник вказує, у скільки разів збільшена швидкість зчитування інформації з оптичного диска порівняно зі стандартною швидкістю 1x (150 Кб/сек.). Стандартна швидкість читання інформації для DVD-дисків (1x) складає 1385 Кб/с. Слід наголосити на існуванні комбайнів – пристроїв, які зчитують інформацію з CD-ROM, записують інформацію на CD-R, перезаписують інформацію на CD-RW, зчитують інформацію з DVD. При використанні пристроїв читання інформації з оптичних дисків з великим множником (більше 40x) слід пам'ятати про небезпеку використання “піратських” дисків – унаслідок високої швидкості обертання погано відцентровані диски “вибухають” у пристрої. Уламки пластику можуть заклинити двигуни, а шматочки металевої фольги закоротити лінії передачі електричних сигналів і напруг живлення. Якщо внаслідок такого “вибуху” вийде з ладу лише блок живлення, то можете вважати, що вам пощастило!



### 3.2.3. Пристрої архівування інформації

При невеликих обсягах інформації (до кількох десятків мегабайт), яку потрібно зберегти, можна використовувати гнучкі магнітні диски. При більших обсягах архівної інформації часто використовують CD-RW. Можливо також використання різних моделей пристроїв ZIP (гнучкі Бернуллі-диски, місткість 35-230 Мб), пристроїв Jazz (змінні жорсткі диски, місткість 1Гб).



Рис. 3.38. Зовнішній вигляд комбінованого пристрою читання інформації з ZIP-дисків і флеш-карт

При великих обсягах архівної інформації (десятки гігабайт) використовують стримери – пристрої запису інформації на магнітну стрічку. Залежно від використовуваних касет та стандартів місткість касет змінюється від сотень мегабайт до десятків гігабайт. Стримери випускають як у внутрішньому, так і в зовнішньому виконанні. Для приєднання стримерів використовують такі інтерфейси:

- інтерфейс ГМД (застаріло);
- інтерфейс ЖМД (IDE);
- інтерфейс SCSI;
- інтерфейс USB;
- оригінальний інтерфейс.

На рис. 3.39 зображені стример та касети для стримерів.



Рис. 3.39. Стример Colorado (внутрішній) та касети для стримерів

### 3.2.4. Пристрої на базі флеш-пам'яті

Пристрої на базі флеш-пам'яті зараз широко використовують як переносні запам'ятовуючі пристрої (флеш-диски) а також у цифрових фото- і відеокамерах (флеш-карти). Основою флеш-пам'яті є польовий транзистор з плаваючим затвором (рис. 3.40).

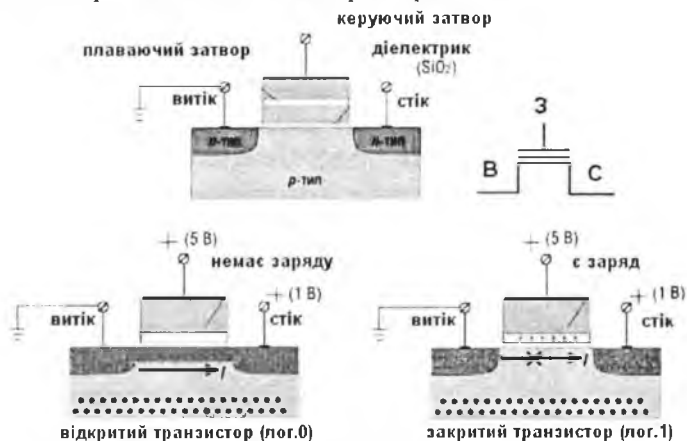


Рис. 3.40. Будова транзистора з плаваючим затвором і читання вмісту комірки пам'яті

За наявності електронів у плаваючому затворі електричне поле, зумовлене ними, витискає носії заряду із призатворної області. Опір переходу "стік-витік" є великим, відповідно спад напруги на цьому переході також (логічна 1). Коли у плаваючому затворі електронів немає, то опір переходу "стік-витік" є малим. Унаслідок цього спад напруги на переході "стік-витік" також є малим (логічний 0). Для уведення зарядів у плаваючий затвор використовують інжекцію гарячих електронів (рис. 3.41,а), або тунелювання електронів Фаулера-Нордхейма (рис. 3.41,б).

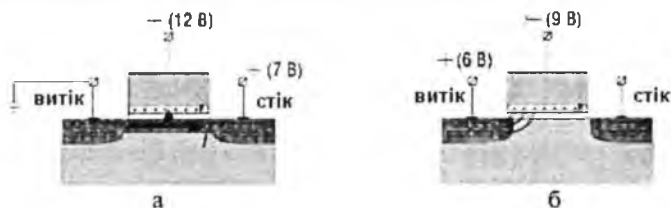


Рис. 3.41. Запис біта інформації методом інжекції гарячих електронів (а) і стирання біта методом квантового тунелювання (б)

Для уведення електронів у плаваючий затвор до затвора і стоку транзистора прикладаються додатні напруги (затвор  $+12\text{В}$ , стік  $+7\text{В}$ ). Під дією цих напруг електрони у каналі  $p$ -типу отримують достатню кінетичну енергію для подолання потенційного бар'єру, зумовленого тонким шаром діелектрика між каналом і затвором (запис логічної 1). Для витиснення електронів із плаваючого затвора до затвора прикладають від'ємну напругу ( $-9\text{В}$ ), а до витоку – додатню ( $+6\text{В}$ ). Це приводить до того, що електрони тунелюють з області плаваючого затвора в область витоку (квантове тунелювання Фаулера-Нордхейма).

Кількість електронів, внесена у плаваючий затвор, визначає величину відпирання переходу “стік-витік”. Цю обставину використовують при побудові багаторівневих комірок. При читанні двох бітів (00, 01, 10, 11), записаних в одну комірку, потрібно визначати один із чотирьох можливих рівнів відкриття переходу “стік-витік”. Для читання 4 бітів, записаних в одну комірку, потрібно розрізняти 16 рівнів відкриття переходу “стік-витік”. Комірки MLC активно розробляються компанією Intel, тому ця технологія отримала назву Intel *StrataFlash*. Для ізоляції транзистора з плаваючим затвором від ліній запису/читання інформації використовують двотранзисторні комірки.

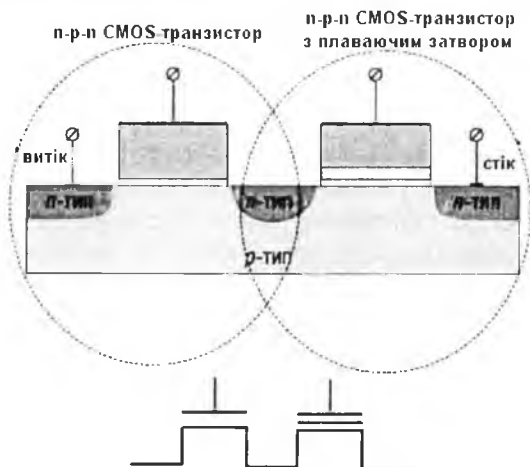


Рис. 3.42. Двотранзисторна комірка пам'яті і її позначення

Перевага двотранзисторних комірок пам'яті полягає в тому, що за їх допомогою можна створювати більш компактні і добре масштабовані мікросхеми пам'яті, оскільки у цьому випадку транзистор з плаваючим

затвором ізолюється від лінії запису/читання інформації. Окрім того, в двотранзисторних комірках пам'яті як для запису інформації, так і для її читання використовують метод квантового тунелювання Фаулера-Нордхейма, що дозволяє знизити напругу запису інформації. Окрім двотранзисторних комірок широко використовують комірки SST (рис. 3.43), розроблені компанією Silicon Storage Technology, Inc. В SST-комірці керуючий затвор вирівняний своїм краєм з краєм стоку, а його вигнута форма дає можливість розташувати плаваючий затвор частково під ним і одночасно над областю витоку. Таке розташування плаваючого затвора дозволяє, з одного боку, спростити процес внесення у нього зарядів методом інжекції гарячих електронів, а з іншого боку – спростити процес витискання зарядів із плаваючого затвора методом тунелювання Фаулера-Нордхейма.

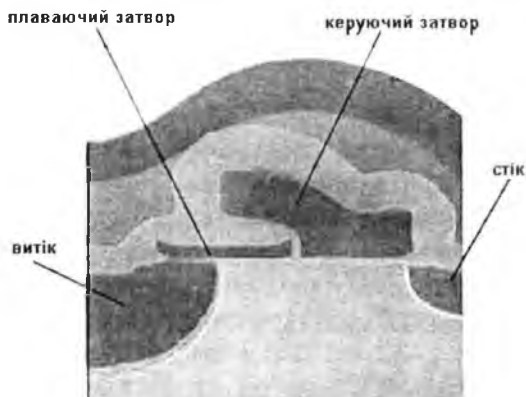


Рис. 3.43. Структура SST-комірки

При витисненні зарядів із плаваючого затвора тунелювання електронів здійснюється на керуючий затвор. Для цього на керуючий затвор подають високу додатню напругу. Під дією електричного поля, створеного цією напругою, відбувається тунелювання електронів з плаваючого затвора на керуючий затвор. Для внесення електронів на плаваючий затвор стік заземлюють, а на витік і керуючий затвор подають додатню напругу. Внаслідок цього між стоком і витоком виникає провідний канал, у якому під дією електричного поля між стоком і витоком електрони набувають кінетичну енергію, достатню для подолання потенціального бар'єра і тунелювання на плаваючий затвор.

Для збереження великої кількості інформації комірки об'єднують у масиви (рис. 3.44-45).

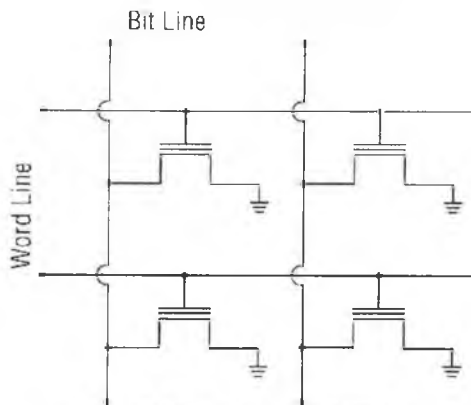


Рис. 3.44. Архітектура NOR

В архітектурі NOR всі затвори під'єднані до лінії керування (Word Line). Аналіз вмісту комірки (0 чи 1) здійснюється по рівню напруги на стоку транзистора, тому всі стоки під'єднані до лінії бітів (Bit Line). Архітектура NOR забезпечує швидкий доступ до пам'яті, однак процеси запису (методом інжекції гарячих електронів) і стирання інформації (квантове тунелювання Фаулера-Нордхейма) відбувається доволі повільно. Окрім того, внаслідок технологічних особливостей виробництва мікросхем флеш-пам'яті з архітектурою NOR, розмір комірки виходить доволі великим. В архітектурі NAND транзистори під'єднують до лінії бітів не по одному (як в архітектурі NOR), а послідовними серіями (рис. 3.45).

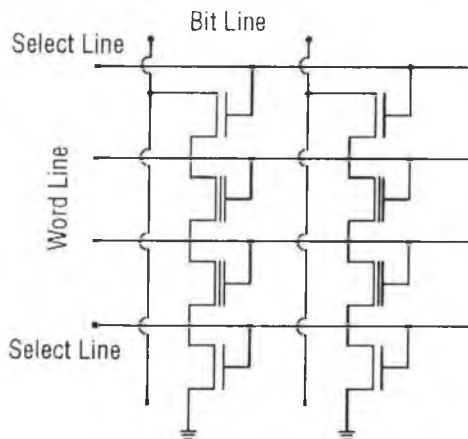


Рис. 3.45. Архітектура NAND

Архітектура NAND дозволяє отримати більш компактне розташування транзисторів, ніж архітектура NOR. Запис інформації здійснюється методом квантового тунелювання, що дозволяє отримати менший час запису, ніж в архітектурі NOR. Для подолання низької швидкості читання інформації мікросхеми NAND обладнують внутрішнім кешом. Окрім розглянутих архітектур NOR і NAND використовують й інші архітектури, наприклад AND, DiNOR тощо, однак вони не отримали масового поширення.

На базі мікросхем флеш-пам'яті продукують флеш-диски з інтерфейсом USB (USB- диски) і флеш-карти для використання у КПК, цифрових фото- і відеокамерах, мультимедійних проекторах, МР3-програвачах і т. ін. Відповідно до конструктиву розрізняють флеш-карти різних форматів:

- Secure Digital (SD) (розміри  $32 \times 24 \times 2,1$  мм, 9 контактів);
- Memory Stick (MS);
- MultiMediaCard (розміри  $24 \times 32 \times 1,4$  мм, RS-MMC –  $24 \times 18 \times 1,4$  мм, 7 контактів);
- xD-Picture Card (розміри  $20 \times 25 \times 1,7$  мм, 22 контакти);
- CompactFlash (CF);
- SmartMedia (розміри  $45 \times 37 \times 0,76$  мм, 22 контакти).



Рис. 3.46. Флеш-карти формату CompactFlash  
(розміри: Type I –  $43 \times 36 \times 3,3$  мм, Type II –  $43 \times 36 \times 5,3$  мм, 50 контактів)



Рис. 3.47. Флеш-карти формату Secure Digital



Рис. 3.48. Флеш-карти формату Memory Stick  
(розміри: – Memory Stick 21,5 x 50 x 2,8 мм, Memory Stick Duo 20 x 31 x 1,6 мм)

Для перенесення інформації з флеш-карт у системний блок ПЕОМ використовують спеціалізовані пристрої читання інформації.

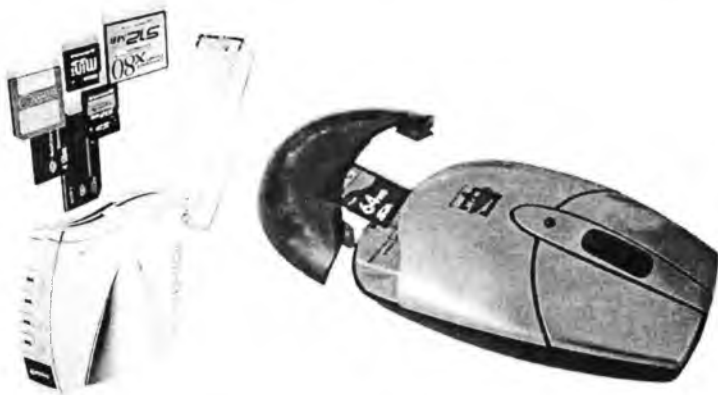


Рис. 3.49. Пристрій читання інформації з флеш-карт

Для перенесення інформації між ПЕОМ широко використовують флеш-диски з інтерфейсом USB.



Рис. 3.50. Флеш-диски з інтерфейсом USB

Оскільки інтерфейс USB дозволяє “гаряче” під’єднання, то робота з флеш-диском дуже подібна до роботи з гнучкими магнітними дисками – під’єднайте флеш-диск до роз’єкта USB (деякі корпуси мають 1-2 роз’єкти цього інтерфейсу на передній панелі) і копіюйте файли з/на флеш-диск, як на звичайний магнітний диск. Невисока ціна флеш-дисків призвела до того, що деякі виробники ПЕОМ (наприклад Apple, Dell) починають відмовлятися від пристроїв читання/запису інформації на ГМД у стандартних конфігураціях.

### Контрольні запитання

1. Як здійснюється запис інформації на магнітний диск?
2. Як здійснюється читання інформації з магнітного диска?
3. У чому полягає відмінність перпендикулярного методу запису від поздовжнього?
4. Як здійснюється читання інформації з оптичного диска?
5. Як здійснюється запис інформації на оптичний диск одноразового запису?
6. Як здійснюється запис інформації на оптичний диск багаторазового запису?
7. Чим відрізняються диски DVD від дисків CD?
8. Які пристрої використовують для архівування інформації?
9. Назвіть сфери застосування флеш-пам’яті.
10. Що таке Raid-масиви і де їх використовують?



### 3.3. Інші пристрої вводу-виводу

#### 3.3.1. Пристрої друку

За конструкцією пристрої друку поділяють на:

- голкові (матричні);
- струменеві;
- електрографічні (лазерні та світлодіодні).

Голкові пристрої друку формують зображення на папері (текстове чи графічне) за допомогою тонких голок, які, вдаряючи по фарбуючій стрічці, формують на папері зображення, що складається з окремих крапок. Назва “матричні” пояснюється тим, що зображення знаків (букв і цифр) формується у прямокутній області (матриці крапок). Кількість голок у різних моделях пристроїв друку різна (9, 18, 24 голки). Чим більша кількість голок, тим вища якість видрукованого символу. Інтерфейс приєднання до системного блоку – Centronics. На рис. 3.51 подано зображення голкового пристрою друку і друкуючої голівки.

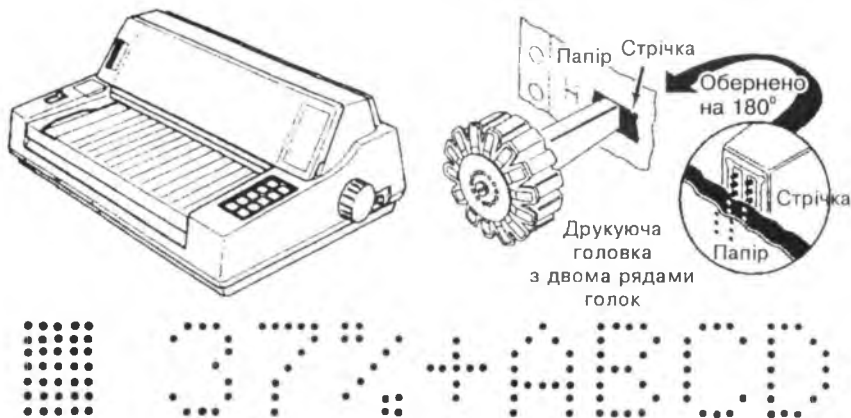


Рис. 3.51. Голковий пристрій друку, друкуюча голівка і зразки синтезу символів з матриці крапок (7 × 5)

Крім голкових пристроїв друку, в яких друкуюча голівка рухається вздовж паперу, існують так звані “лінійні” пристрої – голки розташовані в одну лінію вздовж листа паперу, що дозволяє значно підвищити швидкість друку. Голкові пристрої друку є найдешевшими в експлуатації.

Електрографічні (лазерні та світлодіодні) пристрої друку дають кращу якість зображення, ніж голкові. Для формування зображення оптична система (промінь лазера або промені від лінійки світлодіодів) “прописує” його на електризованому (зарядним коротроном позитивними зарядами) світлочутливому барабані. Лазер або світлодіоди випромінюють світло лише в ті ділянки світлочутливого барабану, які відповідають темним ділянкам зображення. У тих ділянках барабану, на які потрапить промінь лазера (світлодіодів), світлочутливе покриття барабану під дією світла стає провідним і ці ділянки втрачають заряд. Позитивно заряджені зерна фарбуючого порошку прилипають до розряджених ділянок світлочутливого барабану. Передавальний коротрон заряджає папір негативним зарядом, тому при подальшому контакті паперу із світлочутливим барабаном, на якому є зерна фарбуючого порошку, відбувається перенесення зображення на папір. Зображення, сформоване зернами фарбуючого порошку, фіксують на папері шляхом вплавлення його у папір (нагрітий до  $+200\text{ }^{\circ}\text{C}$  закріплюючий валок притискається до паперу з порошком). На рис. 3.52 подана спрощена блок-схема лазерного пристрою друку.

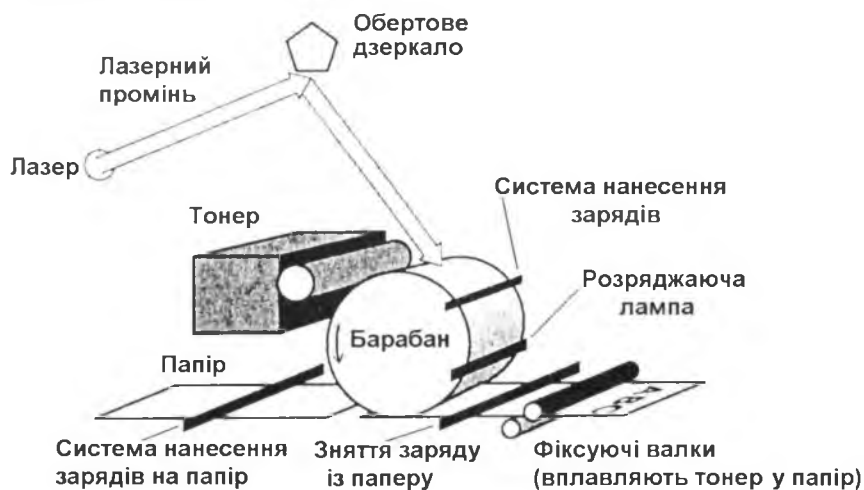


Рис. 3.52. Процес лазерного друку

У світлодіодних пристроях друку світловий потік, який скеровується на світлочутливий барабан, формується лінійкою світлодіодів (рис. 3.53). Всі інші процеси відбуваються як у лазерному пристрої друку.

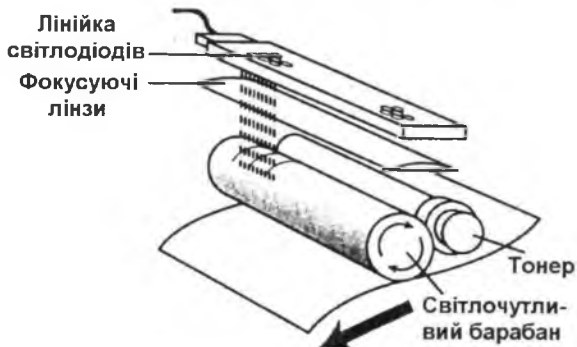


Рис. 3.53. Світлодіодний пристрій друку

У кольорових електрографічних пристроях друку зображення формуються перенесенням на папір чотирьох фарбуючих порошків (чорного і трьох основних кольорів). На рис. 3.54 подано зображення кольорового однопрохідного електрографічного пристрою друку фірми OKI, а на рис. 3.55 – чотирипрохідного фірми Hewlett Packard.



Рис. 3.54. Пристрій кольорового друку OKI C5000: 1 – картриджі із фарбуючими порошками (чорним, голубим, яскраво-червоним і жовтим)



Рис. 3.55. Бюджетний чотирипрохідний пристрій кольорового друку  
HP Color LaserJet 2550

Основні характеристики електрографічних пристроїв друку:

- основний формат паперу – A2, A3 чи A4;
- роздільна здатність Dot Per Inch – dpi – задає кількість видруктованих крапок на дюйм у вертикальному та горизонтальному напрямках (наприклад: 150 × 150 dpi, 300 × 300 dpi, 600 × 600 dpi, 1200 × 600 dpi, 1200 × 1200 dpi, 2400 × 1200 dpi і т. д.);
- швидкість друку (стор/хв);
- рекомендована кількість сторінок на місяць (при якій не спостерігається прискорене зношування механіки пристрою друку);
- ресурс картриджа (у сторінках);
- собівартість однієї сторінки.

Інтерфейс приєднання до системного блоку – Centronics, USB.

Експлуатаційні витрати (папір, фарбуючий порошок, картридж) при використанні електрографічних пристроїв друку вищі, ніж при використанні голкових пристроїв друку. Електрографічні пристрої друку потребують якісного паперу.

Струменеві пристрої друку формують зображення випорскуванням з друкуючої голівки маленьких крапельок фарби – чорної та/або трьох основних кольорів. Залежно від конструкції, пристрій друку має один картридж (з чорною фарбою та з трьома кольорами), два картриджі (один з чорною фарбою, інший – з трьома кольорами) або чотири картриджі (по одному на кожний колір). Крапельки чорнила виштовхуються

через сопла з малим діаметром (близько одного мікрона). Кількість сопел залежить від моделі пристрою друку (від 21 до 128 на один колір). Для виштовхування чорнила з сопла використовують п'єзoeлектричний ефект або тиск пари при швидкому нагріві чорнила мікроскопічним нагрівачем. На рис. 3.56 подані зображення п'єзoeлектричної і термоелектричної голівки.

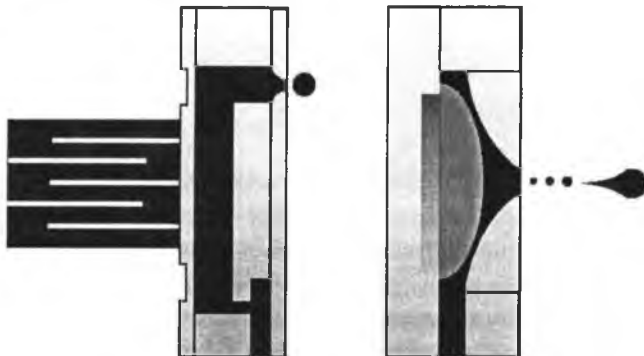


Рис. 3.56. П'єзoeлектрична і термоелектричної голівки струменевого пристрою друку.

Для друку фотографій, отриманих цифровими фотокамерами, використовують струменеві пристрої друку із розширеним набором кольорів (голубим, яскравочервоним, жовтим, світлоголубим, світлочервоним і чорним). На рис. 3.57 подано фотографію струменевого пристрою друку Epson Stylus Photo RX500 і його картриджа.



Рис. 3.57. Струменевий пристрій фотодруку

Окремо слід наголосити на струменевих пристроях друку, які використовують тверді “чорнила” (Хсгох) та пристроях друку з термопереносом, які як барвник використовують швидкозастигаючу пластмасу. На основі струменевої технології створені широкоформатні (до 5м), багатоколірні (8-12) пристрої друку, які використовують у промисловості.

Основні характеристики струменевих пристроїв друку:

- формат паперу – А2, А3, А4 чи інший;
- роздільна здатність;
- швидкість друку (стор/хв);
- рекомендована кількість сторінок на місяць (при якій не спостерігається прискорене зношування механіки пристрою друку);
- ресурс картриджа (у сторінках);
- собівартість однієї сторінки.

Інтерфейс приєднання до системного блоку – Centronix, USB.

Струменеві пристрої друку потребують якісного паперу.

### 3.3.2. Сканери

Сканери використовують для вводу графічних зображень у ПЕОМ. Світловий потік, відбитий від документа, приймається лінійкою фотоприймачів, які перетворюють інтенсивність відбитого потоку у цифровий код. Отримані значення цифрового коду по інтерфейсу передаються у системний блок. Сканери розрізняють за методом переміщення лінійки фотоприймачів уздовж документа:

- ручні – сканер притискається до документу і переміщується вздовж нього вручну;
- рулонні – документ вставляють у приймальну щілину, транспортний механізм протягує документ уздовж лінійки фотоприймачів (завдяки невеликим розмірам використовують з переносними ПЕОМ);
- планшетні – лінійка фотоприймачів переміщується вздовж нерухомого документа;
- слайд-сканери (професійна апаратура) – слайд у спеціальному тримачі переміщується повз оптичну систему з джерелом світла (з одного боку слайда) та лінійкою фотоприймачів (з іншого боку);
- барабанні сканери (професійна апаратура) – негативи прикріплюють до барабана спеціальним гелем і, при обертанні барабана, освітлюють добре сфокусованим світловим променем. Промінь, який пройшов через негатив, потрапляє на фотоприймач.

Деякі планшетні сканери комплектують додатковими пристосуваннями для сканування плівок (у т.ч. фотоплівок).

На рис. 3.58 подана спрощена будова планшетного сканера

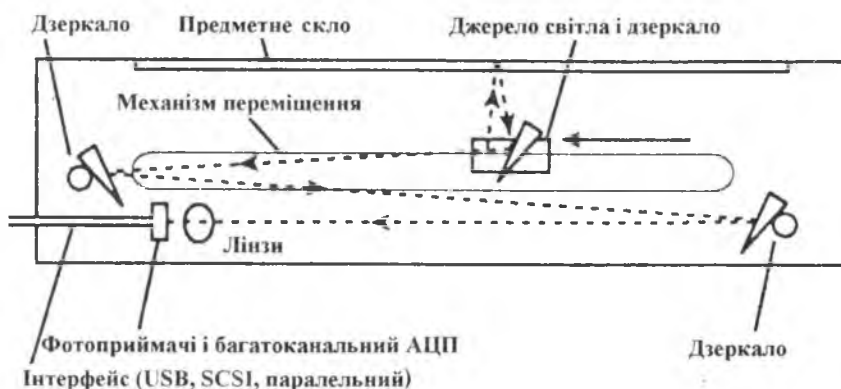


Рис. 3.58. Планшетний сканер

За фотоприймач використовують ПЗЗ-лінійку (див. 3.3.3) та лінійки з мікроскопічними фотодіодами – провідність фотодіода залежить від інтенсивності падаючого на нього світла. Для освітлення документа у сканерах з ПЗЗ-лінійкою використовують люмінесцентні та газорозрядні лампи (рис. 3.59), у сканерах з лінійкою фотодіодів освітлення документа здійснюється лінійкою світлодіодів з керованим кольором світіння (рис.3.60).

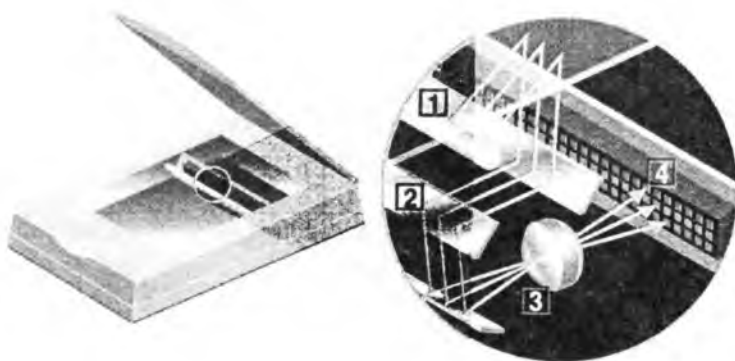


Рис. 3.59. Сканер з лампою білого світла:

1 – лампа білого світла, 2 – дзеркала, 3 – об'єктив, 4 – лінійка фотоприймачів

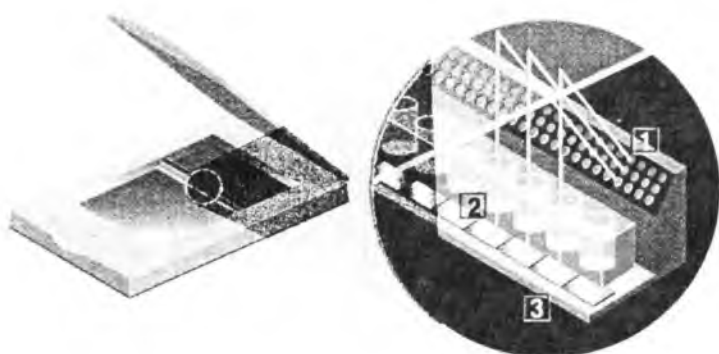


Рис. 3.60. Сканер з лініями світло- і фотодіодів:

1 – лінійка світлодіодів, 2 – стрижневі лінзи, 3 – лінійка фотоприймачів

Основні характеристики сканерів:

- роздільна здатність (dpi) – задає кількість відсканованих точок на дюйм у горизонтальному та вертикальному напрямках (наприклад 1200 x 600 dpi). Роздільна здатність у горизонтальному напрямку визначається кількістю фотоприймачів на лінійці, у вертикальному – кроком переміщення лінійки. Для збільшення роздільної здатності використовують методи інтерполяції (програмна роздільна здатність);
- кількість біт на точку – задає кількість відтінків кольорів, які спроможна “відчувати” лінійка фотоприймачів сканера (24, 36, 42).

Швидкість сканування документа істотно залежить від розміру буферної пам’яті сканера та швидкості передавання інформації у системний блок.

Інтерфейс приєднання до системного блоку – Centronics, USB, SCSI.

Деякі виробники пристроїв друку об’єднують в одному корпусі сканер і пристрій друку з можливістю автономної роботи пристрою (сканування + роздрук відсканованого без участі ПЕОМ).

### 3.3.3. Цифрові фотокамери

Цифрові фотокамери використовують для отримання і вводу в ПЕОМ фотографій у цифровому вигляді. Об’єктив фокусує зображення на ПЗЗ-матриці (рис. 3.61) з великою кількістю світлочутливих елементів (кілька мільйонів).



Прилади з зарядовим зв'язком – ПЗЗ-лінійки і ПЗЗ-матриці були винайдені у 1969 р. американцями Б. Бойлом і Дж. Смітом. На рис. 3.62 подана структура ПЗЗ-лінійки.



Рис. 3.61. ПЗЗ-матриця

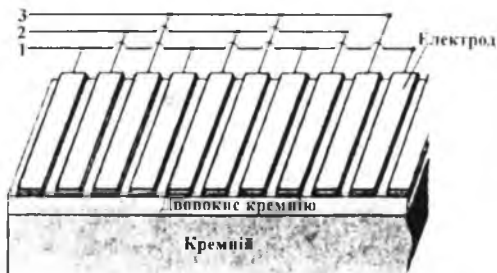


Рис. 3.62. Структура ПЗЗ-лінійки

На кристалі кремнію створюють тришарову структуру, в якій прозорі електроди з полікристалічного кремнію відділені від кристалу кремнію ізолятором (плівкою двоокису кремнію). Кванти світла, які потрапляють у кристал кремнію, створюють у ньому вільні електрони. Завдяки додатній напрузі  $U_{зб}$  (рис. 3.63), прикладеній до шини 2, вільні електрони групуються біля електроду №2 (діелектрична плівка двоокису кремнію не дозволяє електронам перейти на цей електрод).

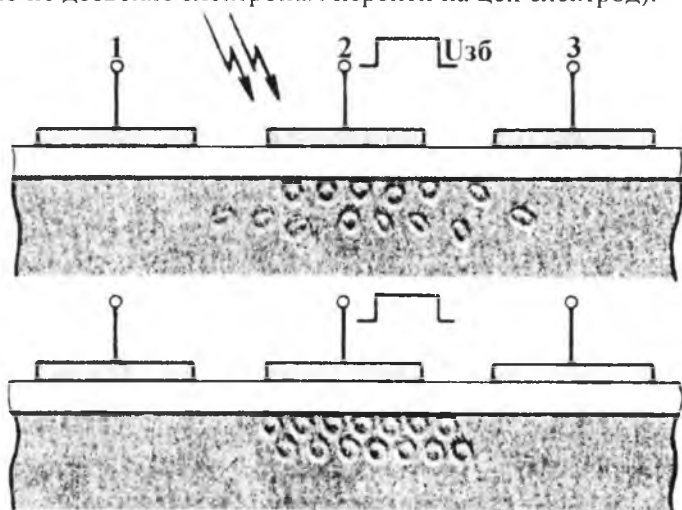


Рис. 3.63. Накопичення електронів у комірки ПЗЗ

Після експозиції світлового потоку на лінійці ПЗЗ до електрода №3 прикладають напругу  $U_{зч}$  ( $U_{зч} > U_{зб}$ ). Електричне поле “перетягує” електрони під електрод №3, після цього напруга  $U_{зб}$  прикладається до цього електрода. Таким чином, перетворення інтенсивності світлового потоку в електричну напругу відбувається таким чином:

1) експозиція – кванти світла, які потряпляють у кристал кремнію, створюють у ньому вільні електрони. Додатня напруга, прикладена до електродів №2, групує вільні електрони під цими електродами;

2) “підрахунок” вільних електронів (читання інформації з лінійки ПЗЗ) – послідовно прикладаючи напруги  $U_{зч}$  –  $U_{зб}$  до електродів №№ 1, 2, 3, вільні електрони “пересувають” до кінця лінійки ПЗЗ. Через *p-n* перехід, створений у кінці лінійки ПЗЗ, вільні електрони стікають на кінцевий електрод, формуючи на навантажувальному опорі імпульс напруги (амплітуда імпульса пропорційна кількості вільних електронів). За допомогою АЦ-перетворювача амплітуду імпульса напруги перетворюють у двійковий код.

Матриці ПЗЗ складаються з певної кількості лінійок ПЗЗ. Зчитування кольорових зображень за допомогою матриць ПЗЗ здійснюють двома способами:

1) вхідний світловий потік за допомогою призми розкладають у спектр. Червону, синю і зелену складові скеровують на окремі ПЗЗ-матриці;

2) на поверхню ПЗЗ-матриці з потроєною кількістю комірок наносять тріади плівкових світлофільтрів (на одному кристалі створюють 3 ПЗЗ-матриці).

Для паралельного виконання експозиції і зчитування інформації на кристалі створюють дві матриці:

- першу використовують для експозиції зображення. Після експозиції зображення вільні електрони комутацією напруг  $U_{зч}$  –  $U_{зб}$  швидко “переганяють” у другу затемнену ПЗЗ-матрицю;

- із другої матриці (матриці зчитування) під час експозиції другого кадру зображення вичитують перший кадр.

Оцифроване зображення можна переглянути на вбудованому у фотокамеру кольоровому рідкокристалічному дисплеї. Для тимчасового зберігання отриманих фотографій фотокамери мають буферну пам’ять, розмір пам’яті (і кількість кадрів) залежать від моделі камери (від 2Мб до 512Мб). Цифрові фотокамери обладнують системами автоматичного

визначення експозиції, автофокусування, запису рухомого зображення, календарем, електронним таймером, фотоспалахом.

Отримані фотографії інтерфейсом RS-232 (Com1, Com2), USB можна передати у системний блок для збереження фотографій на ЖМД.

На рис. 3.64 показано цифрові фотокамери, а на рис. 3.65 – схема приєднання обладнання до цифрової фотокамери Ixus.



Рис. 3.64. Цифрові фотокамери

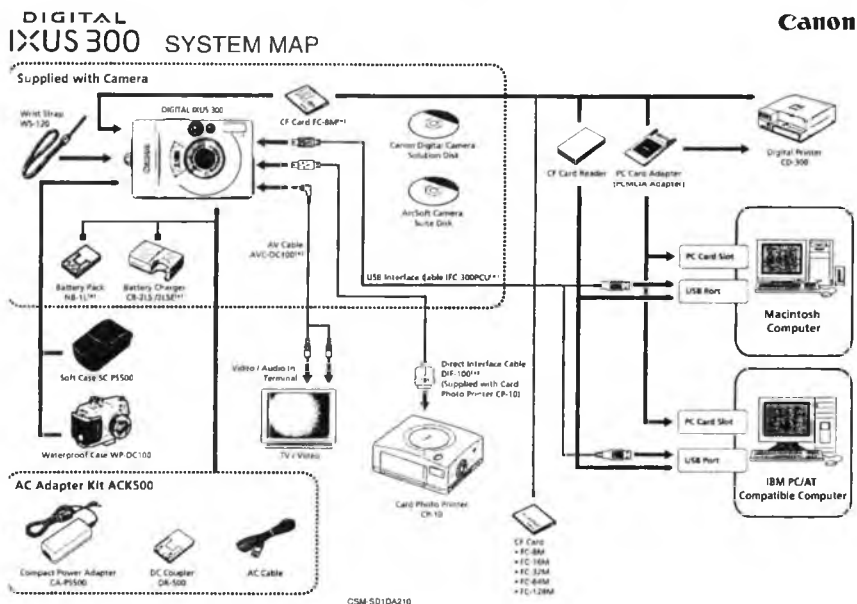


Рис. 3.65. Схема приєднання обладнання до цифрової фотокамери Ixus

### Контрольні запитання

1. Як працює голковий пристрій друку?
2. Як працює струменевий пристрій друку?
3. Як працює електрографічний пристрій друку?
4. Як здійснюється друк кольорових зображень?
5. Які інтерфейси використовують для зв'язку пристроїв друку з системним блоком?
6. Назвіть відомі вам типи сканерів.
7. Як працює планшетний сканер?
8. Які інтерфейси використовують для зв'язку сканерів з системним блоком?
9. Як працює цифровий фотоапарат?
10. Які інтерфейси використовують для переписування цифрових фотографій у системний блок?

## **3.4. Обладнання для локальних та глобальних мереж**

Мережі ПЕОМ використовують для обміну різноманітною інформацією. Залежно від території, яку покриває комп'ютерна мережа, або її протяжності, мережі поділяють на глобальні, міські та локальні.

Глобальні мережі поєднують абонентів у межах країни, континенту чи всієї планети. Тут на першому плані стоїть організація засобів зв'язку. Основним видом зв'язку є супутниковий, радіорелейний або оптоволоконний. Приклади: міжнародна мережа авіакомпаній для замовлення квитків, інформаційна мережа Internet.

Міські мережі поєднують абонентів у межах міста, районів міста, групи закладів чи підприємств, інститутів та ін. Приклади: АСУ міста, інформаційна мережа в межах міністерства, банку.

У глобальній та регіональній мережах основні види робіт – інформаційні. Обчислювальні машини переважно об'єднують не обчислювальні ресурси в динаміці, через мережу, а обмінюються файлами програм і даних. Це потрібно найперше для того, щоб продовжити термін морального старіння ЕОМ і програм, а також отримати доступ до використання розподілених баз даних та організувати швидкий збір статистичних даних. У таких мережах вартість засобів зв'язку більша,

ніж вартість обладнання. Однак термін служби та морального старіння засобів зв'язку значно більший, ніж обладнання ЕОМ (приблизно удвічі).

*Локальні обчислювальні мережі (ЛОМ), або локальні комп'ютерні мережі (ЛКМ), об'єднують ЕОМ, які розташовані на площах, обмежених одним навчальним закладом, підприємством або будівлею. Відстані малі (приблизно 1 кілометр), отже лінії з'єднання дешевші (у кілька разів від регіональних чи глобальних мереж). Тому можна досягнути більших швидкостей передавання інформації, застосовуючи спеціальні лінії передач, наприклад, коаксіальні чи оптоволоконні кабелі. Для порівняння наведемо деякі цифри: швидкості передавання по телефонних лініях становлять 19,2 – 56,0 Кбіт/с, коаксіальних та оптоволоконних – 10-1000 Мбіт/с, тобто на 4-5 порядків більше. Такі швидкості співмірні зі швидкістю зовнішніх шин ПЕОМ. Це дає змогу використовувати ЛКМ не тільки як інформаційну мережу, але й для об'єднання обчислювальних ресурсів. Майже всі ЛКМ є інформаційно-обчислювальними, а їх архітектура є одним з видів архітектури обчислювальної системи. Приклади: навчальні комп'ютерні класи, локальні мережі банківських організацій, податкових служб, офісів.*

### **3.4.1. Загальні принципи побудови обчислювальних мереж**

Незалежно від типу комп'ютерної мережі, існує три типи організації багатоточкового зв'язку між абонентами мережі: комутація ліній, комутація пакетів і комутація повідомлень.

Комутація ліній відбувається за такою схемою:

(ПЕОМ1+модем) ⇔ лінія ⇔ станція ⇔ група комутаційних блоків  
[⇔ лінія ⇔ група комутаційних блоків ⇔ станція]  
⇔ лінія ⇔ (модем+ПЕОМ2)

Недоліком комутації ліній є низька швидкість передавання, високий рівень завад.

Комутація пакетів передбачає постійний зв'язок абонентів. Лінії закріплені за мережею, а інформація передається певними порціями — пакетами. Під *пакетом* розуміємо блок даних фіксованої довжини, який має таку структуру:

- заголовок (службові поля, де зазначені адреси в мережі, пункт джерела пакета, пункт споживача, номер повідомлення та ін.);
- інформаційний блок;
- контрольна сума пакета.

Інформаційне повідомлення розбивають на пакети. Кожний пакет передається незалежно один від одного через найближчий вільний вузол у потрібному напрямку. Пакети можуть проходити через кілька вузлів, а у вузлі призначення знову компонують цілісне повідомлення і передають адресату.

Розглянемо докладніше, як збудована ЛКМ. Головними компонентами є комп'ютери (переважно, це ПЕОМ) та комунікаційне обладнання. Сюди відносять спеціалізовані *контролери* мережі, які вставляють у вільні *роз'язки системної шини*. Це не що інше, як блоки доступу окремого комп'ютера до фізичного середовища передавання даних. Варто зазначити, що об'єднанню підлягають тільки однотипні адаптери. Третій компонент — це лінії зв'язку та роз'язки, що часто називають *інтерфейсом*. І, безумовно, програмне забезпечення.

Тепер розглянемо мережу з погляду її використання. Традиційно ЛКМ розвивалися як засіб розподілу дорогих ресурсів та як засіб керування складними процесами, які відбуваються в режимі реального часу. Основні ресурси, що підлягають розподілу, це дискова пам'ять та пристрої друку. Зрозуміло, що основне навантаження в мережі зосереджується на тих комп'ютерах, які постачають у мережу свої ресурси. Тому комп'ютери поділяють на ті, що постачають свої ресурси в мережу (*сервери*), і ті що споживають їх (*робочі станції*).

Окрім розподілу ресурсів у ЛКМ є важливим розподіл навантаження та розподілене опрацювання даних. До головних функцій локальної мережі належать:

- 1) розподілене використання файлів, тобто багато користувачів отримують доступ до одного файла;
- 2) передавання файлів (швидке копіювання файлів довільних розмірів без застосування дискретних носіїв інформації);
- 3) багатостанційний доступ до інформації та файлів (включаючи запуск прикладних програм з будь-якої станції);
- 4) розподілене використання прикладних програм;

5) одночасне введення даних у прикладні програми (напр., записи у бухгалтерські книги). Ця функція працює тільки зі спеціальними програмами;

6) розподілене використання пристроїв друку;

7) передавання службових повідомлень у текстовому форматі (електронна пошта).

Архітектуру комп'ютерної мережі значною мірою визначає її топологія, тобто спосіб з'єднання ПЕОМ у мережі. ЛКМ будують з комунікаційною підмережею, яка має ідеологію *моноканалу*. Така підмережа може мати різні топології: зірка, кільце, шина, деревоподібна, комутувана та змішана.

1. *Зірка*. Типове розташування абонентів мережі показано на рис. 3.66. Сервер розташовують у центрі, а робочі станції приєднують за допомогою кабелів до нього. Усі робочі станції отримують рівноправний доступ до центрального пристрою. До зіркової топології належать окремі ділянки мережі Fast Ethernet, які будують через приєднання ПЕОМ до комутатора за допомогою кабельних з'єднань.

Переваги: зручне розташування центрального пристрою контролю за станом мережі та збору статистики.

Недоліки: велика протяжність і кількість ліній зв'язку (кожний пристрій приєднується окремим кабелем).



Рис. 3.66. Мережа зіркоподібного типу

2. *Кільце* (циклічна мережа). Спосіб з'єднання ПЕОМ і сервера у мережу показано на рис. 3.67. Передавання даних відбувається за допомогою маркера – спеціальної короткої послідовності сигналів, яка постійно циркулює в кільці. Маркер має властивість приєднувати до

себе інформаційний пакет і транспортувати його до робочої станції за вказаною у його заголовку адресою.



Рис. 3.67. Мережа кільцевого типу, або однорангова кільцева топологія

Недоліком такої мережі є ще й те, що у випадку виходу з ладу кільця на одній ділянці втрачається можливість роботи у всій мережі. Частково цього уникають у модифікованих кільцевих мережах, так званих петлеподібних.

3. Лінійна (шинна) топологія — пряма аналогія магістральних шинних архітектур міні та мікро-ЕОМ. Однією з найпоширеніших мереж лінійної архітектури є мережа Ethernet (Езернет), яку розробили провідні в цій галузі фірми XEROX, а потім DEC та INTEL. Іноді цю мережу позначають DIX. Як приклад, наведемо структуру мережі Ethernet на коаксіальному кабелі (рис. 3.68). Тут мережа складається з чотирьох лінійних ділянок, які називаються сегментами. За допомогою спеціальних пристроїв сегменти під'єднуються до головної шини, утворюючи у такий спосіб деревовидну структуру. Така структура дозволяє будувати багатосерверні мережі, бо кожний з сегментів може мати свій сервер. Характерною особливістю таких мереж є узгодження хвильових опорів інформаційних шин (коаксіальних кабелів) за допомогою спеціальних пристроїв — термінаторів (“заглушок”). У випадку мережі Езернет — це звичайний резистор опором у 50 Ом.



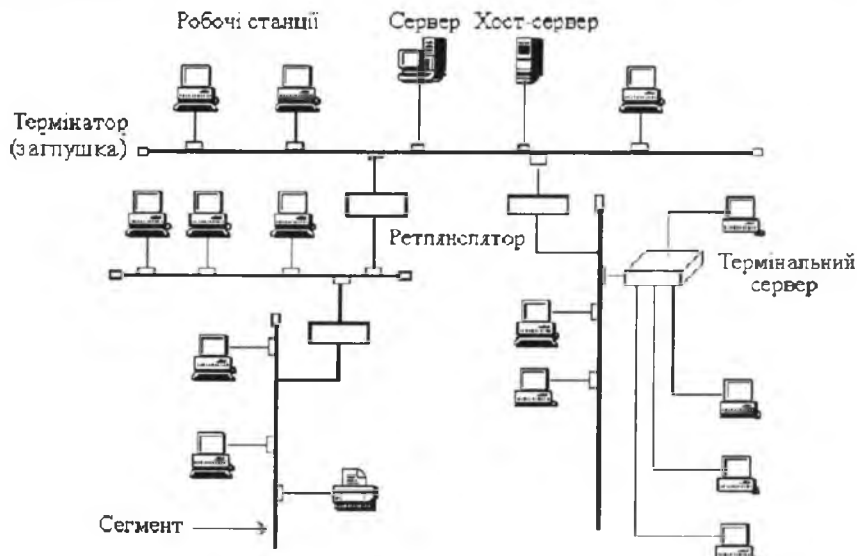


Рис. 3.68. Шинна топологія (Мережа типу “Езернет” на коаксіальному кабелі)



Рис. 3.69. Приклад мережі з використанням комутаторів

3. *Комутувана топологія.* Сучасний спосіб організації приєднання робочих станцій до мережі. Це швидше технологія, ніж топологія, яка полягає у приєднанні робочих станцій до спеціального обладнання – комутаторів. Комутатори можуть з'єднуватись між собою або бути приєднаними до сервера. Розрізняють комутатори для об'єднання ПЕОМ з однотипними адресами (це так звані хаби (англ. Hub)) та з можливістю маршрутизації адрес робочих станцій, тобто об'єднання локальних мереж у мультимережі чи корпоративні мережі.

Звичайно, існують і *складні (змішані) топології*, у яких за допомогою комутаторів чи іншого обладнання поєднують роботу мереж зіркового, кільцевого та інших типів.

### 3.4.2. Пристрої для побудови локальних мереж

Для побудови локальної мережі потрібно мати в кожній ПЕОМ контролер локальної мережі (рис. 3.70) та фізичне середовище передавання електричних сигналів між ПЕОМ. Контролер локальної мережі – це плата розширення для шини PCI або ISA (деякі системні плати мають вбудовані контролери локальної мережі), швидкість передавання інформації – 10, 100 або 1000 Мбіт/с. Фізичне середовище – це система спеціальних кабелів (коаксіальний або так звана “скручена пара”), яка з’єднує між собою всі ПЕОМ. Окрім електричних кабелів використовують оптичні кабелі (досягається висока швидкість передавання інформації), однак вартість такої локальної мережі поки що висока.

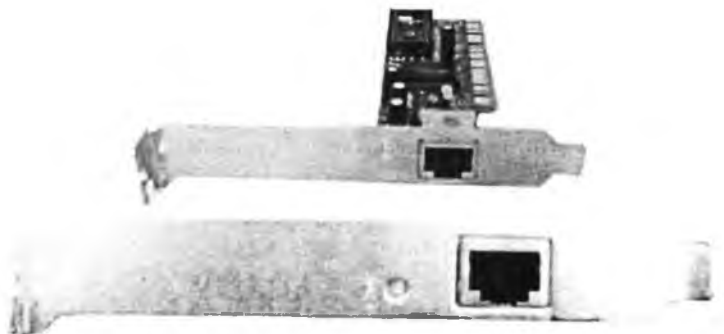


Рис. 3.70. Контролер локальної мережі

На відміну від магістральної структури локальної мережі для побудови зіркоподібної структури потрібно додатково придбати мережевий концентратор (рис. 3.71) або комутатор. Основні характеристики концентратора (комутатора) – це кількість портів для приєднання ПЕОМ (під “скручену пару” + один, варіативно, під коаксіальний кабель) та швидкість передавання інформації. Локальні мережі з комутаторами працюють швидше, але комутатори дорожчі від концентраторів (з однаковою кількістю портів).



Рис. 3.71. Комутатор (згори) і концентратор

Окрім кабеля “скручена пара” для передавання цифрової інформації можна використовувати й інші металеві провідники – хоча б провідники мережі електроживлення 220В (110В), яких є вдосталь у стінах наших приміщень (технологія PowerLine). На рис. 3.72,а подана структура мережі PowerLine, а на рис. 3.72,б – адаптер PowerLine.

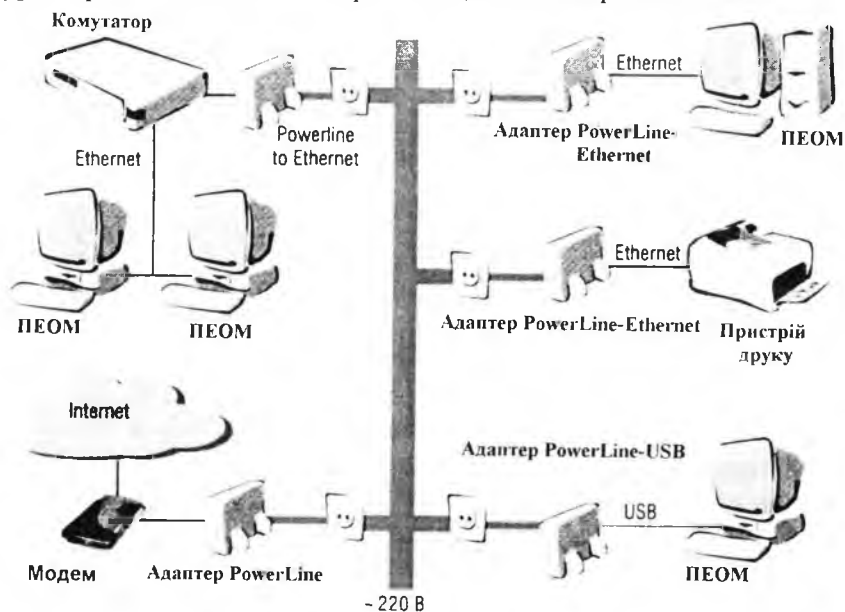


Рис. 3.72,а. Структура мережі PowerLine



Рис. 3.72,б. Адаптер PowerLine

Ця технологія передбачає три діапазони швидкостей передавання інформації:

- низьку – до 0,05 Кбіт/с на відстані до десятків кілометрів;
- середню – від 0,05 до 50 Кбіт/с на відстані до декількох кілометрів (смуга частот 50-535 кГц);
- високу – більше 100 Кбіт/с на відстань до декількох сотень метрів (смуга частот 1,7-30 МГц).

### 3.4.3 Бездротові мережі

Технології бездротового доступу до обладнання можна розділити на три групи:

- бездротові локальні мережі (Wireless Local Area Network – WLAN);
- бездротові персональні мережі (Wireless Personal Area Network – WPAN);
- бездротове передавання/приймання інформації на великі відстані (див. далі).

#### 3.4.3.1. Бездротові персональні мережі

Для побудови бездротових персональних мереж використовують:

- стандарт HomeRF, який працює у діапазоні 2,4 ГГц і забезпечує швидкість передавання інформації від 0,8 до 10 Мбіт/с. Випромінювана потужність не більше 100 мВт;
- стандарт Bluetooth (IEEE 802.15.1), розрахований на смугу частот 2,4-2,4835 ГГц і забезпечує передавання/приймання інформації на невеликі відстані (до 10 метрів) зі швидкістю 1 Мбіт/с. На рис. 3.73

подана схема взаємодії пристроїв цієї мережі, а на рис. 3.74 – адаптер Bluetooth з інтерфейсом USB. Розширення стандарту Enhanced Data Rate (EDR) Bluetooth забезпечує передавання/приймання інформації з швидкістю 2.1 Мбіт/с;

- групу стандартів IEEE 802.15.x. Стандарт IEEE 802.15.3 базується на діапазоні 2,4 ГГц і повинен забезпечувати швидкість передавання інформації від 11 до 55 Мбіт/с. Стандарт IEEE 802.15.4 використовує 16 каналів у діапазоні 2,4 ГГц, 10 каналів у діапазоні 935 МГц і один канал на частоті 868 МГц. Швидкість передавання інформації від 20 до 250 Кбіт/с;

- стандарт ZigBee, який працює в тому ж частотному діапазоні, що й IEEE 802.15.5. Мережа, побудована на цьому стандарті, може мати 254 клієнта і один керуючий (master) пристрій.



Рис. 3.73. Схема взаємодії пристроїв стандарту Bluetooth



Рис. 3.74. USB-адаптер Bluetooth

### 3.4.3.2. Бездротові локальні мережі

Для побудови бездротових локальних мереж використовують:

- групу стандартів IEEE 802.11.x. Стандарт IEEE 802.11a (1999 р.) розрахований на роботу у діапазоні 5 ГГц і забезпечує такі швидкості обміну інформацією – 24, 12, 6 Мбіт/с (окрім перерахованих обов'язкових для всіх пристроїв швидкостей є ще необов'язкові швидкості – 54, 48, 36, 18, 9 Мбіт/с). Стандарт IEEE 802.11b (1999 р.) розрахований на роботу у діапазоні 2,4 ГГц і забезпечує такі швидкості обміну інформацією – 11, 4,5, 2, 1 Мбіт/с. Стандарт IEEE 802.11g (2001 р.) розрахований на роботу у діапазоні 2,4 ГГц з максимальною швидкістю 54 Мбіт/сек. Пристрої стандартів IEEE 802.11b і IEEE 802.11g є сумісними між собою. Пристрої стандарту IEEE 802.11a несумісні з пристроями стандартів IEEE 802.11b і IEEE 802.11g;

- групу стандартів HyperLan. Стандарт HyperLan/1 забезпечує швидкість передавання інформації 23 Мбіт/с, HyperLan/2 – 54 Мбіт/с. Стандарти відрізняються методами частотної маніпуляції.

У стандарті IEEE 802.11.x використовується радіозв'язок між ПЕОМ та локальною мережею на рис. 3.75 подана структура локальної мережі з бездротовим доступом.

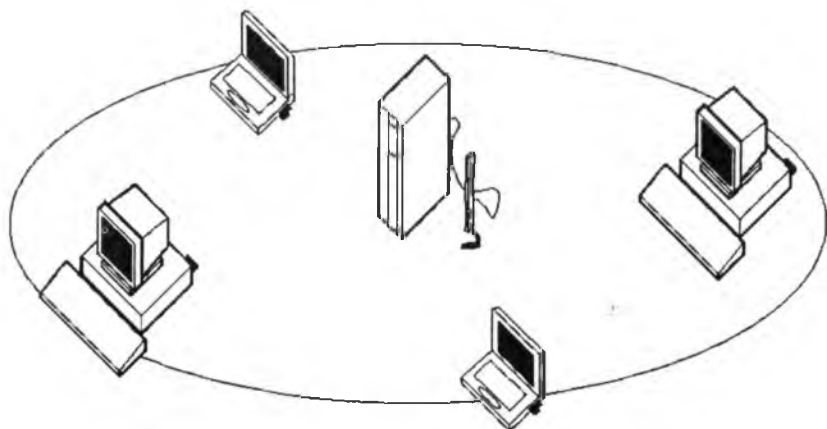


Рис. 3.75. Бездротова мережа ПЕОМ

Бездротовий доступ до мережі за цим стандартом забезпечується завдяки використанню:

- контролерів бездротового доступу (рис. 3.76-3.78)



Рис. 3.76. Контролер бездротового доступу стандарту IEEE 802.11 для переносних ПЕОМ (PC-card):  
а – вбудована антенна, b,c – світлодіоди (b – приймання/передавання інформації, c – живлення), d – гніздо для приєднання зовнішньої антени

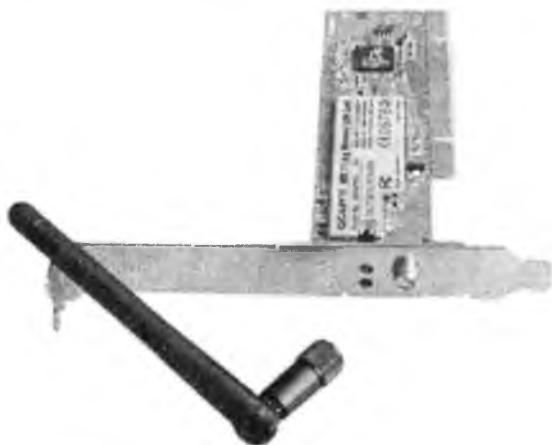


Рис. 3.77. Контролер бездротового доступу стандарту IEEE 802.11 у вигляді плати розширення



Рис. 3.78. USB-контролер бездротового доступу стандарту IEEE 802.11

- *точок доступу* (рис. 3.79-3.80)

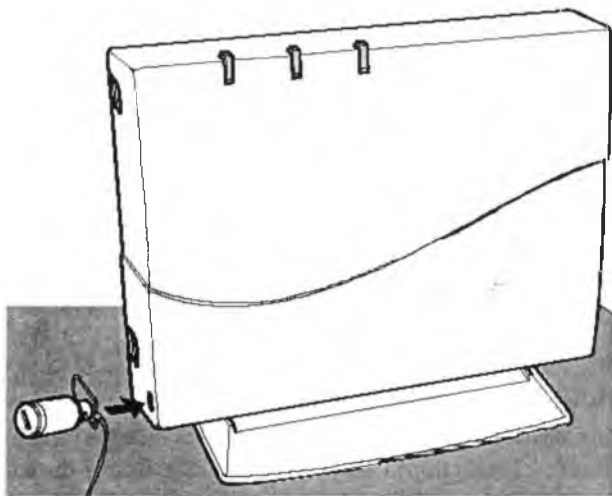


Рис. 3.79. Настільний пристрій бездротового доступу до мережі (Access Point-I)



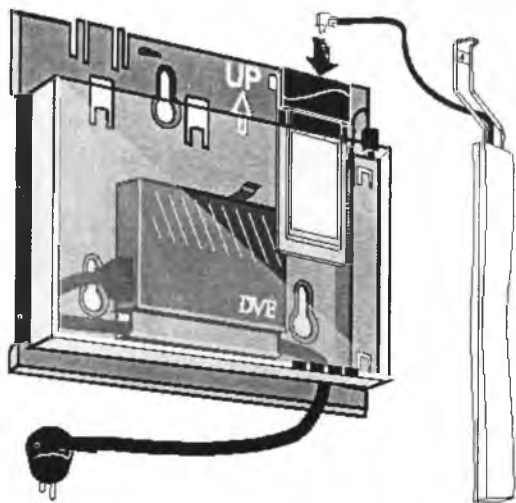


Рис. 3.80. Настінний пристрій бездротового доступу до мережі (Access Point-II)  
 • *зовнішніх антен*

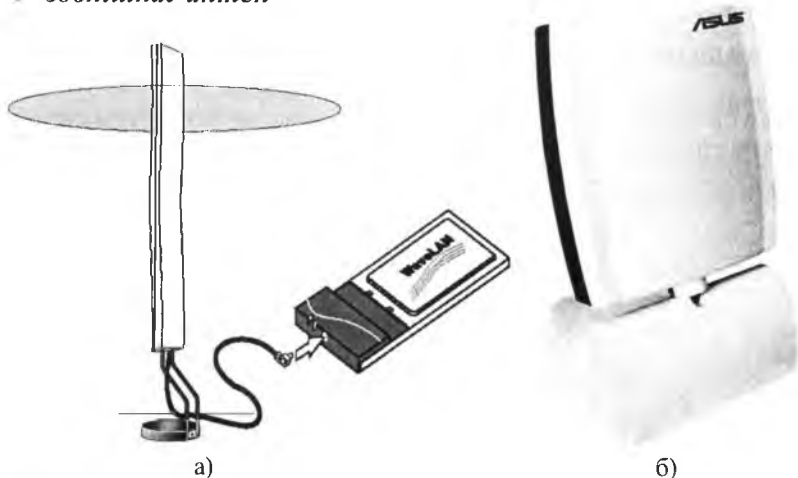


Рис. 3.81. Зовнішні антени (а – з коловою діаграмою, б – скерована)

Останнім часом в Україні активно встановлюють оптичні лінії доступу до локальних мереж, які ґрунтуються на використанні інфрачервоних лазерних світлодіодів потужністю 150 мВт і фотоприймачів. Обладнання оптичних ліній встановлюють на підвіконнях або дахах будівель. Оптичні лінії працюють у межах прямої видимості (до 5 км.).

Вартість обладнання для оптичних ліній доволі висока (кілька десятків тисяч у.о.), однак значно менша від вартості оптоволоконних ліній передавання цифрової інформації. На відміну від систем радіодоступу оптичні системи не створюють радіозавад і не потребують ліцензування.

#### 3.4.4. Пристрої для роботи у глобальних мережах

Побудова глобальних мереж з використанням коаксіальних кабелів або “скрученої пари” технічно й економічно неможлива. Для побудови глобальних мереж використовують інші канали передавання інформації. Одним із таких каналів є телефонна мережа, яка є доволі розгалуженою (у різних країнах по-різному). Для “пристосування” сигналів ПЕОМ до телефонної мережі, яка розрахована на передавання голосових сигналів, використовують спеціальні пристрої – модеми (*МО*дулятор-*ДЕ*Модулятор). Термін модулятор означає, що на передавальному кінці мережі двійкові сигнали перетворюються в електричні коливання звукової частоти (рис. 3.82), а на приймальному – демодулятор розпізнає (демодулює) вид надісланого коливання і перетворює його в двійкові “1” або “0”.

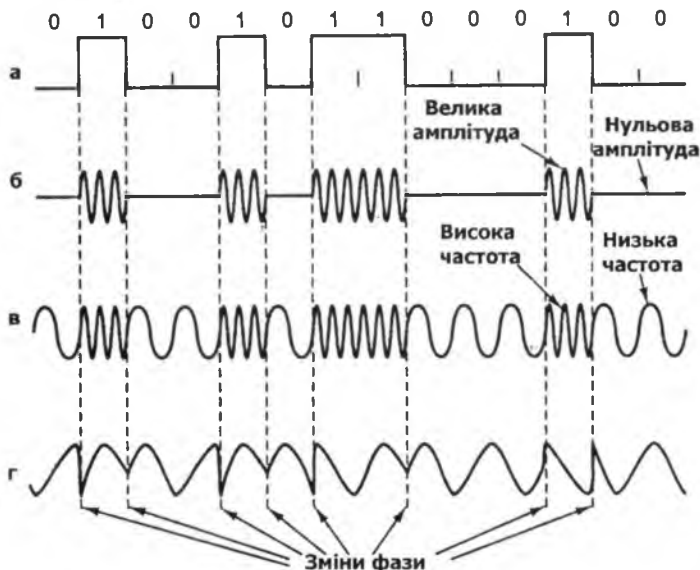


Рис. 3.82. Процес модуляції сигналу, який подається у телефонну лінію

Для перетворення двійкових сигналів в електричні коливання використовують амплітудну маніпуляцію (рис. 3.82,б), частотну маніпуляцію (рис. 3.82,в) і фазову маніпуляцію (рис. 3.82,г). Сучасні модеми, як правило, використовують кілька видів маніпуляції, що дозволяє передавати декілька бітів за один такт передачі. Кількість таких тактів у секунді вимірюють у бодах. Модеми, які передають і приймають інформацію одночасно (за рахунок використання однієї частоти на передавання інформації, іншої – на приймання інформації), називають дуплексними. Модеми, які лише передають або лише приймають інформацію у конкретний момент часу, називають напівдуплексними. Модеми випускають у двох варіантах:

- як плату розширення (внутрішній);
- як незалежний пристрій зі своїм блоком живлення (зовнішній).

Внутрішні модеми поділяють на апаратні (зовнішній без блока живлення і корпусу) та програмні (сигнали, які подаються у телефонну лінію, формує процесор, розпізнавання сигналів з телефонної лінії також виконує процесор). Програмний модем додатково навантажує процесор. Зовнішній модем більш зручний для некваліфікованого користувача завдяки світловій та звуковій індикації роботи модема. Доцільно купувати модем, який дозволяє паралельне приєднання телефонного апарата (на корпусі модема має бути роз'єм для телефонного апарата з позначкою Phone). На рис. 3.83 зображені внутрішні (internal) та зовнішній (external) модеми.



Рис. 3.83. Внутрішні (internal) та зовнішній (external) модеми:  
1 – внутрішній модем для шини PCI, 2 – внутрішній модем у стандарті PC-card,  
3 – зовнішній модем

Вибір модема для українського користувача істотно залежить від характеристик телефонної мережі, з якою він працює. Важливим показником модема є максимальна швидкість передачі інформації, яку він

підтримує. Сучасні модеми забезпечують максимальну швидкість передачі 33600 біт/с за протоколом V.34+ та 56000 біт/с за протоколами V.90, V.92, X2, K56Flex.

Додатковими (і дуже корисними) функціями модема можуть бути:

- можливість приймання/передавання факсів (факс-модем);
- можливість приєднання мікрофона і зовнішнього гучномовця (голосовий модем) – це перетворює модем на телефонний апарат, як зовнішній гучномовець можна використати звукову карту з голосниками.

Приєднання зовнішніх модемів до системного блоку здійснюється через інтерфейси RS-232 та USB.

Передавання інформації від модема до АТС і навпаки здійснюється по телефонних лініях. Можливі два варіанти використання телефонних ліній:

- “комутована лінія” – телефонна лінія використовується і для телефонних розмов, і для передавання інформації. Для доступу в Інтернет ви повинні набрати телефонний номер провайдера, і після встановлення з’єднання з ним почати працювати в Інтернеті. Швидкість передавання інформації не більша 64 Кбіт/с;

- “виділена лінія” – одна (дві) телефонні лінії постійно використовуються лише для роботи в Інтернеті. Швидкість передавання інформації – 36-112 Кбіт/с.

У першому випадку ви сплачуєте вартість користування телефоном за тривалістю роботи телефонної лінії вашій АТС і сплачуєте за послуги провайдера (як правило, за обсягом прийнятої і переданої інформації). У другому випадку ви оплачуєте АТС орендну плату за користування “виділеною лінією”, а послуги провайдера сплачуєте за обсягом прийнятої і переданої інформації. У нічний час і АТС, і провайдери працюють за пільговим тарифом.

Якщо на АТС встановлено спеціальне обладнання для xDSL-зв’язку (Digital Subscrube Line, x – вид DSL) і телефонні лінії мають достатньо високі частотні характеристики, то за допомогою xDSL-зв’язку можна отримати такі швидкості передавання інформації:

ADSL (Asymetrical Digital Subscrube Line) – асиметричний канал. Швидкість передавання вхідної інформації до 8 Мбіт/с, вихідної – до 1 Мбіт/с;

ADSL2 – швидкість передавання вхідної інформації до 12 Мбіт/с, вихідної – до 1 Мбіт/с;

ADSL2+ – швидкість передавання вхідної інформації до 20 Мбіт/с, вихідної – до 1 Мбіт/с;

CDSL (Consumer Digital Subscrube Line) – повільніший канал. Швидкість передавання вхідної інформації 1 Мбіт/с, вихідної – 640 Кбіт/с;

UDSL (Universal Digital Subscrube Line) – швидкість передавання вхідної інформації 1,544 – 6,0 Мбіт/с, вихідної – 128-384 Кбіт/с.

При використанні xDSL-зв'язку можна одночасно працювати в Інтернеті і вести телефонні розмови.

Принцип дії xDSL-зв'язку полягає у розширенні частотного діапазону сигналів, які передаються телефонним кабелем. Завдяки використанню розділюючих частотних фільтрів по телефонній лінії одночасно може передаватись цифровий потік (двоскерований) і телефонна розмова.

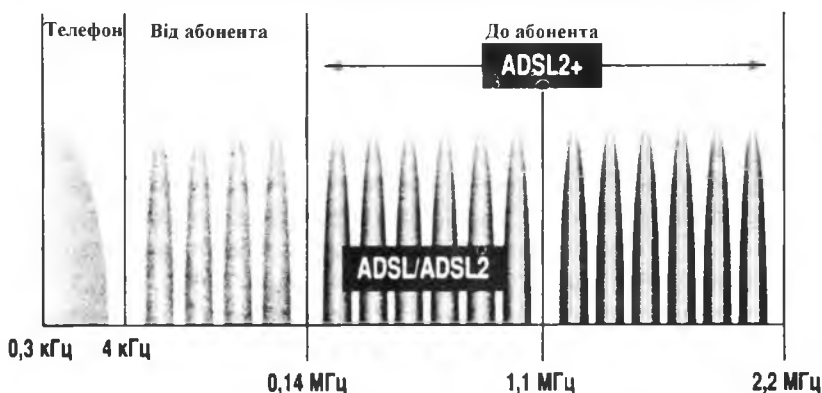


Рис. 3.84. Розподіл частотного діапазону ADSL2+

Максимальна швидкість передавання інформації у лініях xDSL-зв'язку залежить від відстані між абонентом і АТС.

Відстань (км)	Від абонента (Мбіт/с)	До абонента (Мбіт/с)
1,8	1,216	8,126
2,7	1,088	7,872
3,7	0,864	3,648
4,3	0,640	1,984
4,6	0,544	1,408
4,9	0,416	0,960
5,2	0,320	0,576
5,5	0,224	0,320
5,8	0,128	0,128

Для xDSL-зв'язку потрібно укласти договір із організацією (провайдером), яка надає послуги xDSL-зв'язку, і встановити xDSL-модем (деякі провайдери надають модем в оренду). На рис. 3.85 подана структура xDSL-мережі, а на рис. 3.86 – ADSL-модеми.

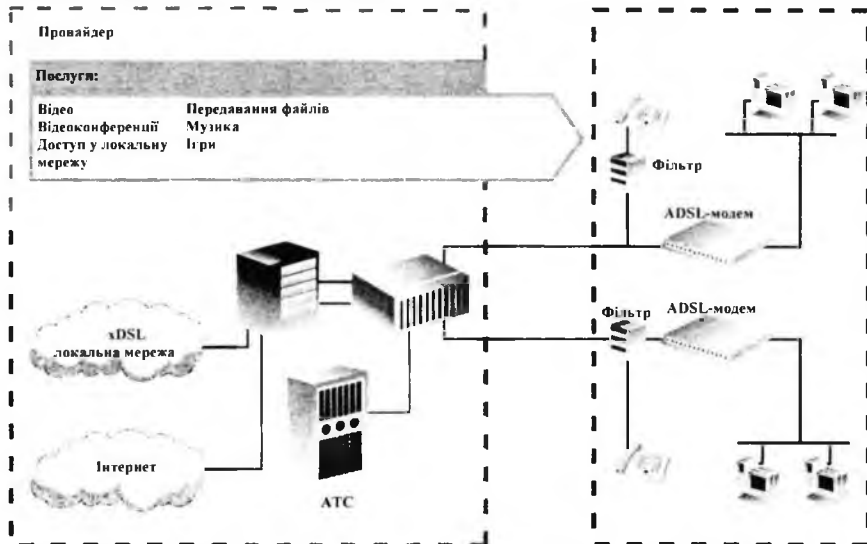


Рис. 3.85. Структура xDSL-мережі



Рис. 3.86. ADSL-модеми

Для роботи в Інтернеті можна використовувати мережі кабельного телебачення (якщо фірма, яка надає послуги кабельного телебачення,

встановила у себе спеціальне високовартісне обладнання). На рис. 3.87 подана схема з'єднання ПЕОМ з кабельною мережею.

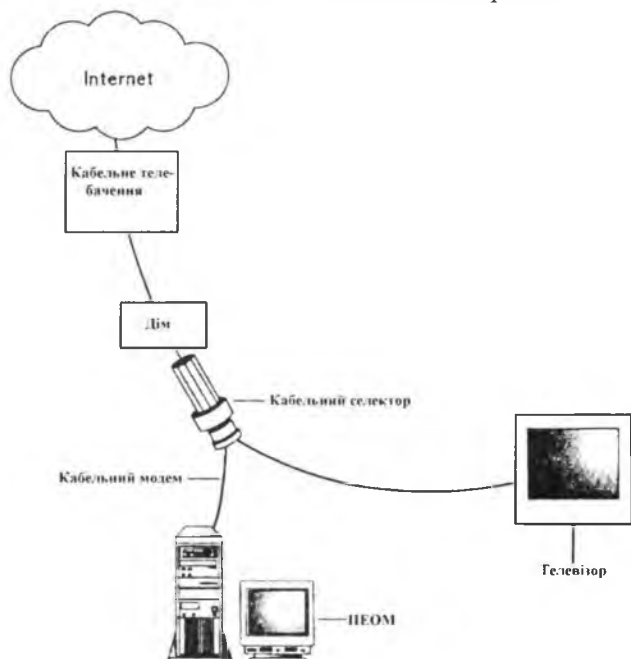


Рис. 3.87. Схема з'єднання ПЕОМ з мережею кабельного телебачення

З'єднання з Інтернетом можливе і по супутниковим каналам, на рис. 3.88 подана схема з'єднання DirecPC.



Рис. 3.88. Схема з'єднання ПЕОМ з використанням DirecPC

При використанні супутникового каналу також можливі два варіанти:

- асиметричний – запит на інформацію передається провайдеру по дротових лініях, інформація з Інтернету – по супутниковому каналу;
- симетричний – і запит на інформацію, й інформація з Інтернету передаються по супутникових каналах.

Для бездротового обміну інформацією між наземними станціями, рознесеними на відстань до 70 км можна використовувати:

- обладнання стандарту IEEE 802.11x із зовнішніми скерованими антенами;
- обладнання стандарту IEEE 802.16a, який базується на діапазоні 2-11 ГГц і забезпечує швидкість до 70 Мбіт/с.

Методом бездротового доступу до Інтернету також є використання мобільних телефонів зі стандартом GPRS (General Packet Radio Service). Для з'єднання з ПЕОМ використовують інтерфейси RS-232, USB, Bluetooth, IrDA. Максимальна швидкість передачі інформації – 171,3 Кбіт/с.

### 3.4.5. Обладнання для IP-телефонії

Якщо модеми і мобільні GPRS-телефони є засобами для доступу в Інтернет, то обладнання для IP-телефонії дозволяє використовувати локальні і глобальні мережі для голосового зв'язку. IP-зв'язок між абонентами полягає у передаванні оцифрованих і стиснутих голосових сигналів (розрізаних на пакети) по мережах, призначених для передавання цифрової інформації. На рис. 3.89 подана структура IP-зв'язку між двома офісами фірми, які можуть розташовуватись в різних країнах.



Рис. 3.89. Структура IP-зв'язку між віддаленими офісами фірми



IP-шлюз об'єднує в один цифровий потік цифрові сигнали від ПЕОМ та аналогові сигнали від АТС, факсів і аналогових телефонів при передаванні інформації. При прийманні інформації відбувається зворотня операція розділення цих сигналів. На керуючий сервер провайдера покладено завдання передавання пакетів інформації по глобальній мережі. Вартість IP-зв'язку менша від вартості міжнародного зв'язку.

Окрім обладнання IP-зв'язку через глобальні мережі виробники пропонують цифрові IP-системи для забезпечення голосового зв'язку за допомогою локальних мереж (IP-PBX). Таке обладнання дозволяє відмовитись від прокладання телефонних ліній у приміщеннях. На рис. 3.90 подано обладнання для IP-телефонії.



Рис. 3.90. Обладнання для IP-телефонії:  
згори вниз – IP-шлюз, IP-PBX, IP-телефон

IP-зв'язок можуть використовувати й окремі користувачі через звичайний телефон за участі IP-провайдера ("Голден телеком", "Інфоком", "Пріоком", "Фарлеп", Macrotel та ін.) або у конфігурації "ПЕОМ – ПЕОМ" за умови встановлення на них мікрофонів, голосників (телефонів) і відповідного програмного забезпечення (сервери Dialpad, Go2Call). На рис. 3.91 подані різні варіанти IP-зв'язку.

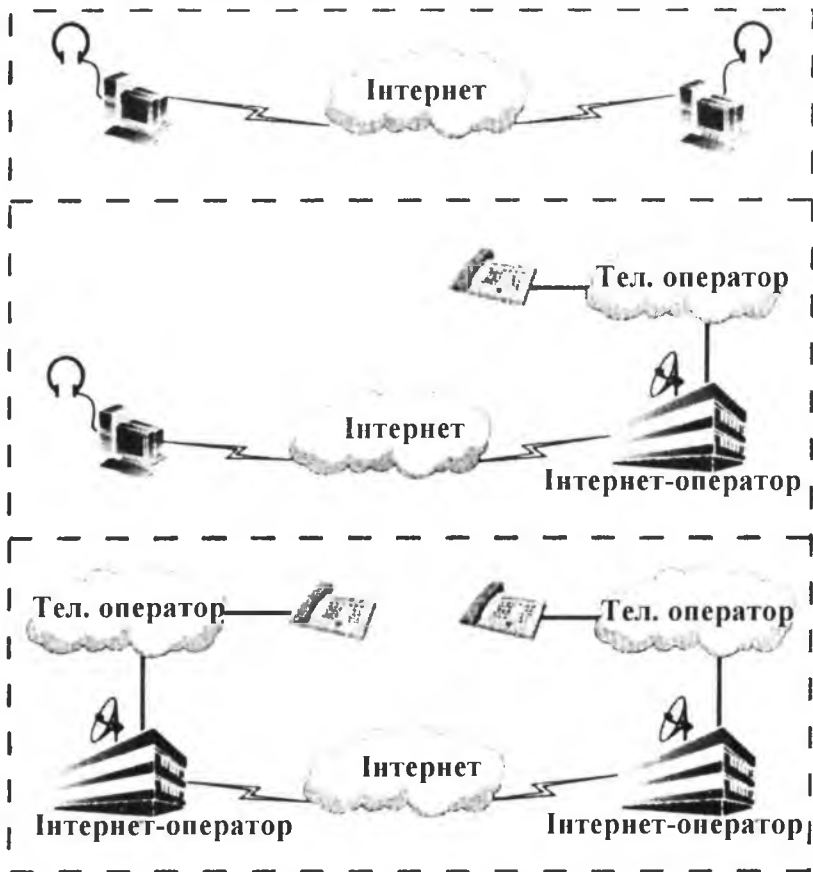


Рис. 3.91. Варіанти IP-зв'язку

### Контрольні запитання

1. Які засоби використовують для побудови локальних мереж?
2. Перерахуйте відомі вам методи доступу до глобальних мереж.
3. Поясніть призначення і принципи роботи модема.
4. Що таке xDSL-зв'язок?
5. Що таке IP-телефонія? Назвіть області застосування IP-телефонії.

## 3.5. Засоби Multimedia

### 3.5.1. Звукові карти

Звукові карти використовують для відтворення/запису фонограм. Останнім часом виробники широко пропонують системні плати з інтегрованим виконанням звукового каналу, що де-факто перетворює його у стандартний інтерфейс (цифровий звуковий канал). Цифровий звуковий канал забезпечує запис і відтворення моно- та стереозвуку. Робота звукового каналу підтримується відповідним програмним забезпеченням (драйверами), які постачаються разом з обладнанням. На рис. 3.92 зображена планка з роз'ємними звукового каналу.

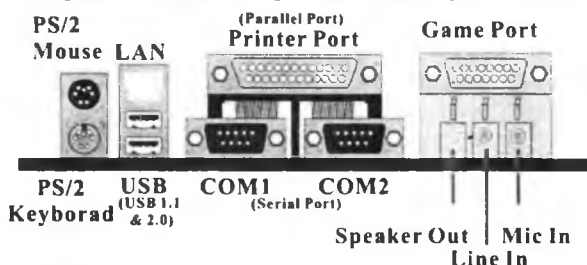


Рис. 3.92. Роз'ємня звукового каналу

Для забезпечення об'ємного (3D) звучання за допомогою голосників, виробники пропонують звукові карти з виходом на три, чотири, п'ять або шість голосників. На рис. 3.93 подані фотографії сучасних звукових карт.

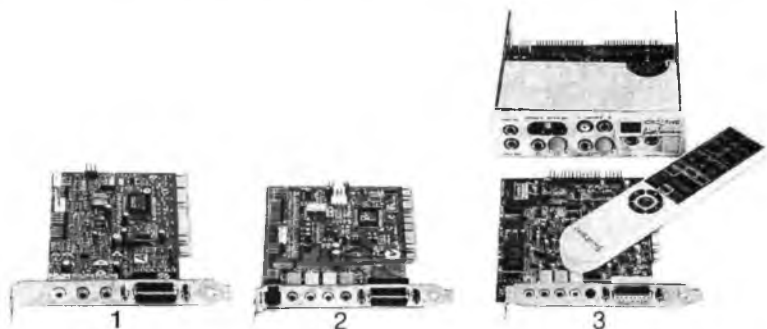


Рис. 3.93. Сучасні звукові карти (об'ємне звучання):

- 1 – звукова карта з виходом на 2 голосники, 2 – звукова карта з виходом на 3 (2+1) голосники, 3 – звукова карта з виходом на 5 (4+1) голосників, 5,25" відсіком з цифровими й оптичними виходами і пультом дистанційного керування

Слід звернути увагу на появу зовнішніх пристроїв уведення/виведення звуку, які під'єднуються до ПЕОМ по інтерфейсу USB.



Рис. 3.94. Пристрій введення/виведення звуку з інтерфейсом USB

### 3.5.2 Засоби запису і прослуховування звуку

Голосники (звукові колонки) та навушники використовують для відтворення звукового супроводу програм (у першу чергу ігрових), відеофільмів, фонограм (при прослуховуванні на ПЕОМ звукових оптичних дисків), радіопередач (за наявності FM-приймача), телепередач (за наявності TV-приймача). Споживацькі характеристики засобів відтворення звуку змінюються у широких межах. Корпуси голосників (звукових колонок) з внутрішнього боку покривають магнітопровідним матеріалом для зняття впливу магнітного поля працюючих голосників на зображення на екрані дисплея. Для створення ефекту об'ємності звуку використовують кілька голосників (3-5). На рис. 3.95 подана схема розміщення п'яти голосників для забезпечення об'ємного звучання.

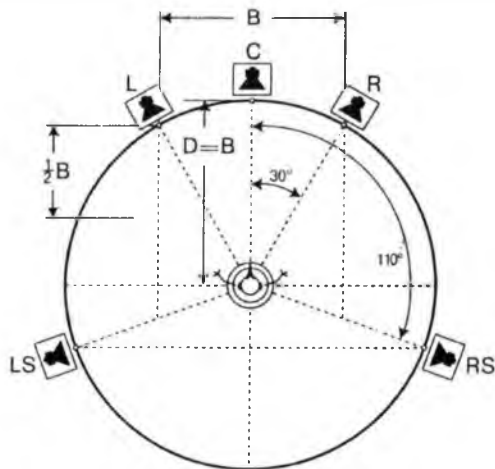


Рис. 3.95. Схема розміщення голосників

Якщо ви маєте високоякісний Hi-Fi підсилювач і колонки, ви можете під'єднатися до лінійного виходу “звукової карти”. Для запису фонограм використовують мікрофон і відповідний вхід цифрового звукового каналу, а також лінійний вхід (для будь-якого джерела сигналу розмахом до 200 мВ).

На рис. 3.96 показано голосники (звукові колонки), головні телефони і мікрофони, які пропонуються комп'ютерними фірмами.

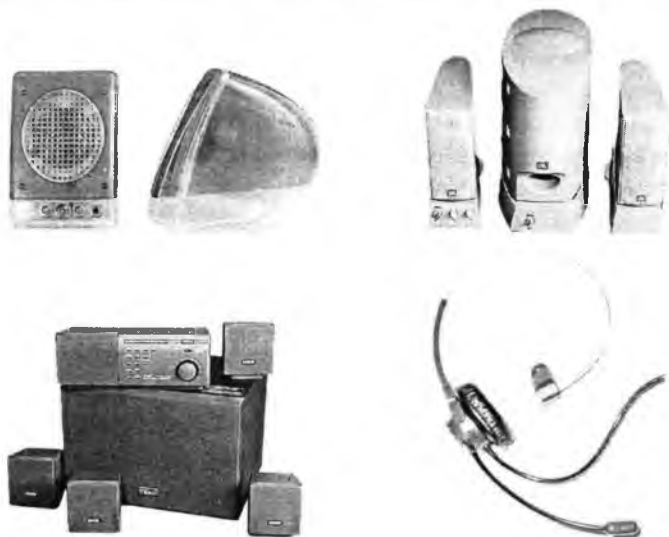


Рис. 3.96. Голосники (звукові колонки), навушники (головні телефони) і мікрофони

### 3.5.3. Системи віртуальної реальності

Системи віртуальної реальності (СВР) використовують як у професійній сфері (тренажери, для прийняття стратегічних рішень, навчальні системи), так і для розваг (інтерактивне кіно, атракціони, ігри). Найпростішим засобом СВР є стереоокуляри, які дозволяють переглядати стереозображення на екрані дисплея. Деякі виробники графічних контролерів додають стереоокуляри до комплекта поставки. Віртуальні шоломи об'єднують в одному пристрої стереотелефони, два мікро-дисплеї (по одному на око) та (варіативно) давач повороту голови – Head-Tracker, завдяки якому забезпечується зміна зображення при повороті

голови. Кіберрукавички дозволяють маніпулювати віртуальними об'єктами та "відчувати їх на дотик". Перед використанням стереоокулярів та віртуальних шоломів варто проконсультуватись у лікаря – люди з слабкою психікою можуть мати проблеми!

Ігрові маніпулятори, доповнені пристроями, які імітують реакцію модельованого механізму на дії оператора (наприклад вібрації керма мотоцикла на поганій дорозі) та максимально подібні до органів керування реальних пристроїв (штурвал літака, кермо мотоцикла, кермо+газ+гальмо+перемикач швидкостей), утворюють ще одну групу пристроїв СВР.

### 3.5.4. Web-камери

Для проведення відеоконференцій у межах одного приміщення (за умов наявності локальної мережі) та в різних приміщеннях (за умов наявності швидкісного Інтернету) використовують Web-камери, які встановлюють біля кожної ПЕОМ. Використання Web-камер, голосників та мікрофонів дозволяє провести наради між рознесеними у просторі співрозмовниками з прослуховуванням їх повідомлень та спогляданням їх міміки.

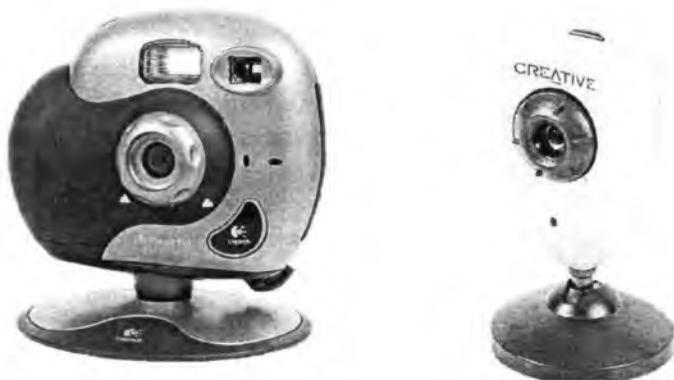


Рис. 3.97. Зовнішній вигляд Web-камер

Web-камери складаються з об'єктива, ПЗЗ-матриці (див. цифрові фотокамери) і контролера, який керує роботою Web-камери і передає отримане зображення у системний блок по інтерфесу USB. Деякі виробники пропонують Web-камери із бездротовою передачею інформації, на їх основі можна будувати системи відеоспостереження (див. рис. 3.98).



Рис. 3.98. Web-камера із бездротовою передачею інформації

### 3.5.5. Приймачі FM- та TV-сигналів

Для прослуховування радіопередач в ультракороткохвильовому діапазоні з частотною модуляцією можна використовувати звукові карти та графічні контролери з FM-приймачем. Такі карти комплектуються спеціальною програмою (драйвером), яка забезпечує програмне керування приймачем. Для перегляду телепередач можна придбати TV-приймач (внутрішній або зовнішній), більшість таких пристроїв комплектується пультом дистанційного керування для вибору каналу та зміни гучності/яскравості/контрасту.

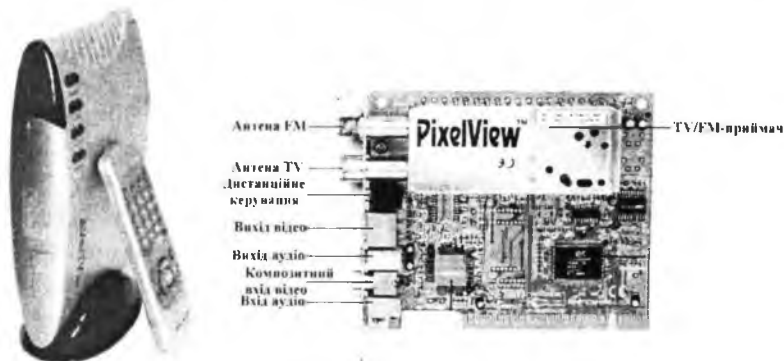


Рис. 3.99. Зовнішній і внутрішній TV-приймач

Деякі графічні контролери мають додаткові функції — ввід і вивід телевізійних сигналів (TV-in, TV-out), про що згадувалось у 3.1.2. У поєднанні з програмами редагування телевізійних кадрів їх використовують для монтажу відеофільмів (титри, відео ефекти, звуковий супровід і т.ін.)

### Контрольні запитання

1. Які засоби використовують для відтворення звуку на ПЕОМ?
2. Як створюється об'ємне звучання на ПЕОМ?
3. Які засоби використовують для запису звуку на ПЕОМ?
4. Що таке системи віртуальної реальності?
5. Поясніть принципи роботи WEB-камери. Назвіть відомі вам застосування WEB-камер.
6. Які засоби забезпечують приймання радіосигналів у FM-діапазоні на ПЕОМ?
7. Які засоби забезпечують приймання телепрограм на ПЕОМ?

## 3.6. Встановлення ПЕОМ

При встановленні ПЕОМ слід враховувати такі обставини:

- розташування дисплея навпроти вікна призводить до накладання світлового потоку з вікна на світловий потік від екрану дисплея, що спричинює підвищення навантаження на очі користувача;
- увімкнені дисплеї з електроннопроменевою трубкою створюють змінне електромагнітне поле (від системи розгортки електронного променя), яке має масимуми напруженості ззаду і з боків дисплея;
- верхня поверхня столу повинна мати розміри 90 см у глибину та 90-120 см у ширину (для забезпечення необхідної глибини стола деякі фірми пропонують столи з висувною полицею для клавіатури і мишки). Наступний рисунок ілюструє вимоги до правильного розміщення ПЕОМ.

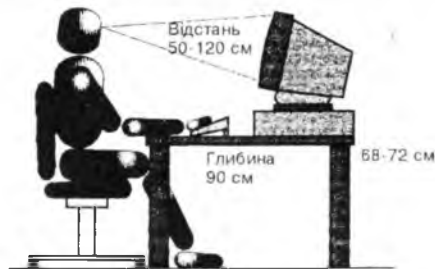


Рис. 3.100. Рекомендоване розміщення ПЕОМ на столі



Подальші дії по встановленню ПЕОМ можуть бути такі:

- 1) встановіть у легкодоступному місці мережевий фільтр (рис. 2.117) і під'єднайте його до електромережі 220В;
- 2) встановіть на столі системний блок, дисплей, клавіатуру, мишку (пристрій друку, сканер, голосники і т. ін.);
- 3) під'єднайте мережеві (220В) шнури до цих пристроїв та до мережевого фільтра;
- 5) під'єднайте до системного блока кабелі від дисплея, мишки, клавіатури (та інших пристроїв, якщо ви їх придбали), на рис. 3.101 зображена задня стінка системного блока з усіма роз'ємними;
- 5) увімкніть мережевий фільтр (індикатор фільтра засвітиться червоним кольором);
- 6) знайдіть кнопки "Power" на системному блоці і дисплеї, увімкніть системний блок і дисплей натисканням на ці кнопки;
- 7) дочекайтеся закінчення процесу завантаження операційної системи. Бажаємо успіхів у праці на ПЕОМ!

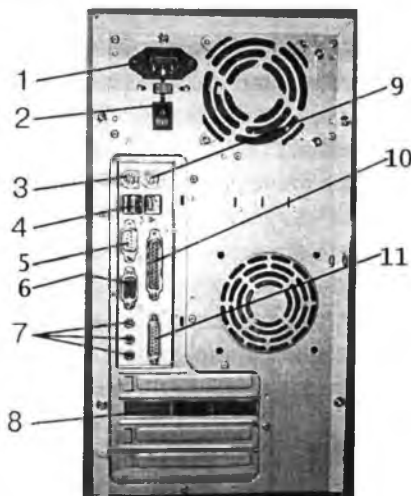


Рис. 3.101. Роз'єми системного блоку для приєднання периферійних пристроїв:

- 1 – гніздо для мережевого (220В) кабеля, 2 – вимикач блока живлення (ATX) системного блока, 3 – роз'єм для клавіатури (порт PS/2), 4 – роз'єм для інтерфейсу USB, 5 – роз'єм для послідовного інтерфейсу Com1, 6 – роз'єм для кабелю дисплея, 7 – входи і виходи звукового каналу, 8 – роз'єм для контролера локальної мережі, 9 – роз'єм для мишки (порт PS/2), 10 – роз'єм для пристрою друку, 11 – роз'єм для ігрового порта

---

## СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

---

1. Аладьев В. З., Хунт Ю. Я., Шишаков М. Л. Основы информатики: Учеб. пособие. – М., 1999.
2. Биркгоф Г., Барті Т. Современная прикладная алгебра / Пер. с англ. – М., 1976.
3. Борзенко А. Е. IBM PC: устройство, ремонт, модернизация. – М., 1997.
4. Брамм Д., Брамм П. Микропроцессор 80386 и его программирование. – М., 1990.
5. Буров Є. Комп'ютерні мережі. – Львів, 1999.
6. Горбунов В. Л., Панфилов Д. И., Преспухин Д. Л. Справочное пособие по микропроцессорам и микроЭВМ. – М., 1988.
7. Григорьев В. Л. Микропроцессор i486. Архитектура и программирование (в 4-х книгах). – М., 1993.
8. Гук М. Процессоры Intel: от 8086 до Pentium II. – СПб., 1997.
9. Гук М. Аппаратные средства IBM PC. Энциклопедия – СПб, 1999.
10. Дао Л. Программирование микропроцессора 8088. Пер. с англ. – М., 1988.
11. Злобін Г. Г. Посібник користувача персонального комп'ютера: Навч. посіб. – Львів, 2003.
12. Енциклопедія кібернетики (у 2 томах) – К., 1977.
13. Каган Б. М. Электронные вычислительные машины и системы: Учеб. пособие для вузов. – М., 1991.
14. Колодницький М. М. Технічне та програмне забезпечення комп'ютерних інформаційних технологій. – Житомир, 1995.
15. Ланцов А. Л., Зворыкин Л. Н., Осипов И. Ф. Цифровые устройства на комплементарных МПД интегральных микросхемах. – М., 1983.
16. Локажук В. М. Мікропроцесори та мікроЕОМ у виробничих системах: Посібник. – К., 2002.
17. Морс С. П., Алберт Д. Архитектура микропроцессора 80286. – М., 1990.
18. Мюллер С. Модернизация и ремонт ПК: Учеб. пособ. – 11-е изд. Пер. с англ. – М., 2000.
19. Мячев А. А., Степанов В. Н. Персональные ЭВМ и микроЭВМ. Основы организации: Справочник. – М., 1991.
20. Рикалюк Р. Є., Стягар О. М., Данчак П. В. Вступ до комп'ютерних мереж: Текст лекцій. – Львів, 1996.
21. Рикалюк Р. Є. Архітектура комп'ютерів: Текст лекцій. – Львів, 2002.

22. *Смит Б. Э., Джонсон М. Т.* Архитектура и программирование микропроцессора Intel 80386.
23. *Смирнов А. Д.*, Архитектура вычислительных систем. Учеб. пособие. – М., 1990.
24. *Степаненко О. С.* Досье пользователя: компьютер внутри. – К., 1998.
25. *Таненбаум Э.* Архитектура компьютеров. – 2002.
26. Толковый словарь по вычислительным системам / Под ред. В. Иллиnguорта. – М., 1991.
27. *Фигурнов В. Э.* IBM PC для пользователя. Краткий курс. – 7-е изд. – М., 1997.
28. *Хэмминг Р. В.* Теория кодирования и теория информации / Пер. с англ. – М., 1983.
29. *Шеннон К.* Работы по теории информации и кибернетике / Пер. с англ. – М., 1963.
30. 386™SX Microprocessor programmer's reference manual. – Intel Corporation, 1989.
31. i486™ Processor programmer's reference manual. – Intel Corporation, 1990.
32. *Micael Edelhart.* Intel's official guide to 386 Computing. – Osborne McGraw-Hill, 1991.
33. *Neal Margulis.* i860™ Microprocessor Architecture. – Intel, 1990.

## **ДОДАТОК 1. МЕТОДИ ПОШУКУ НЕСПРАВНОСТЕЙ У ПЕОМ**

Якщо гарантійний термін на ПЕОМ ще не скінчився (один-три роки), то при виникненні несправностей доцільно звернутися на фірму, яка продала ПЕОМ. Однак перед тим варто спробувати визначити тип несправності. Перевірку справності зовнішніх (по відношенню до системного блоку) пристроїв можна визначити заміною їх на справні з іншої ПЕОМ (дисплей, клавіатура, мишка). Несправність блоку живлення системного блоку можна встановити по відсутності свічення світлодіода “Power” (+5В) і нерухомості вентилятора блоку живлення (+12В), однак варто переконатися, що змінна напруга 220В надходить на блок живлення – перевірте розетку і шнур живлення. Якщо блок живлення справний, зверніть увагу на повідомлення тесту POST – звукові і текстові. За їх допомогою можна приблизно визначити несправну складову системного блоку. Остаточне визначення несправності і заміну несправної частини виконає фірма, яка продала вам ПЕОМ.

Коли ж гарантійний термін скінчився, то вам залишається або визначити несправну складову самостійно, придбати її і встановити на ПЕОМ, або ж відвезти несправний системний блок (дисплей, пристрій друку, сканер, модем, комутатор) на комп’ютерну фірму й оплатити ремонт (несправні клавіатуру або мишку варто просто замінити).

При самостійному визначенні несправної складової дійте уважно і без поспіху. Якщо технічна документація на складові ПЕОМ у вас збереглася, ознайомтеся з нею. У деяких випадках причиною несправності є “перегорання” запобіжника у блоці живлення системного блоку (дисплею, пристрою друку, сканера, модема, комутатора). Якщо ви вмієте користуватись омметром, спробуйте знайти несправний запобіжник і замінити його. На корпусі запобіжника вказується струм спрацювання 2А, 3А і т. ін. Не встановлюйте запобіжники з більшим струмом спрацювання! Якщо після заміни запобіжника несправність не ліквідовано, перевірте, чи блок живлення виробляє потрібні напруги. Для цього до нього достатньо під’єднати ЖМД і виміряти вольтметром напруги +5В, +12В на роз’єдні для живлення 5,25” пристроїв. Ремонт блоку живлення системного блоку або зовнішнього блоку живлення сканера, модема, комутатора може коштувати дорожче, ніж придбання справного.

Коли ви маєте доступ до кількох (хоча б двох) однотипних ПЕОМ, можна скористатися справною ПЕОМ для визначення несправної складової. Визначення несправної складової системного блока можна здійснити за таким алгоритмом:

1) відкрийте системний блок і від'єднайте від блока живлення всі пристрої, крім системної плати;

2) акуратно зніміть плати розширення, залишивши лише графічний контролер;

3) під'єднайте до графічного контролера справний дисплей і увімкніть дисплей і системний блок;

4) проаналізуйте повідомлення тесту POST (один довгий, три коротких гудки – несправний графічний контролер, періодично повторювані гудки – несправна системна плата, ніяких повідомлень [ні звукових, ні текстових] – несправна системна плата, короткий перелік цифрових кодів помилок подано у п.5.2.1.5.1);

5) якщо несправна плата розширення графічного контролера – замініть його і почніть перевірку спочатку. Якщо графічний контролер інтегрований (встановлений на системній платі), виставте відповідну опцію SETUP у стан Disable і встановіть плату розширення графічного контролера;

6) якщо несправна системна плата, перейдіть до перевірки її складових (див. далі)

7) якщо ж системна плата і графічний контролер справні, поступово під'єднуйте від'єднані складові (при встановленні плат розширення контролюйте надійність контакту плати розширення з гніздом розширення [ISA, VLB, PCI, AGP]). Таким чином ви виявите несправну складову (плату розширення, ЖМД, ГМД).

Несправності системної плати можна розділити на такі групи:

- несправності вбудованих контролерів (див. далі);
- несправність модулів пам'яті – замініть модулі пам'яті;
- несправність мікропроцесора – замініть мікропроцесор;
- несправність охолоджувача мікропроцесора – замініть охолоджувач;
- несправність стабілізатора живлення мікропроцесора – замініть стабілізатор (починаючи з мікропроцесора I80486DX2/80 напруга живлення є меншою 5В). Якщо мікросхему стабілізатора замінити не вдається – замініть системну плату.

Якщо на системній платі несправний контролер клавіатури – замініть системну плату, коли ж несправні контролери стандартних інтерфейсів (послідовного, паралельного, ГМД, ЖМД) – встановіть відповідні опції SETUP у стан Disable і встановіть плату розширення портів вводу/виводу (Multi Input/Output – MIO).

Далі подані деякі типові проблемні ситуації і рекомендації щодо їх усунення.

Після ввімкнення живлення на дисплеї з'являється напис

***Sorry, but this disk is not bootable.***

***Insert SYSTEM disk into the drive.***

***OK***

Можливі причини:

- 1) у пристрої роботи з ГМД знаходиться дискета, з якою ви працювали (витягніть її і натисніть будь-яку клавішу);
- 2) вийшов з ладу жорсткий системний диск (запустіть програму Fdisk або програму тестування ЖМД);
- 3) неправильно задані параметри ЖМД (див. 3.1.5.2).

Після ввімкнення живлення на дисплеї з'являється напис

***System disk failure, insert system disk and press Enter***

Можливі причини:

- втрата інформації про параметри ЖМД (відновіть її за допомогою програми Setup або задайте опцію Auto)
- втрачені системні файли на ЖМД, з якого завантажується операційна система, або несправний ЖМД (спробуйте завантажитись з аварійної дискети та перегляньте параметри ЖМД програмою Fdisk (*будьте обережні при роботі з Fdisk, неакуратне використання її може призвести до втрати всієї інформації на ЖМД*). Якщо програма Fdisk підтвердить наявність справного ЖМД у системному блоці, відновіть системні файли на ЖМД і перезавантажте ПЕОМ. Коли ж програма Fdisk повідомить про відсутність ЖМД – звертайтеся до фахівця.)

Після ввімкнення живлення ПЕОМ на дисплеї з'являється напис

***Keyboard error or keyboard not present***

Можливі причини:

- кабель клавіатури витягнутий з роз'єкта на системному блоці, або має поганий контакт з роз'єктом (поправте роз'єкт, але якщо для приєднання клавіатури використовується порт PS/2, то операції приєднання/від'єднання можна робити лише при вимкненому живленні!);
- несправна клавіатура (замініть клавіатуру, для перевірки можна скористатись іншою справною клавіатурою);
- несправний контролер клавіатури на системній платі (замініть системну плату).

*Після ввімкнення живлення ПЕОМ та дисплею засвічуються світлодіоди на системному блоці, дисплеї та клавіатурі, однак на екрані дисплея відсутнє зображення*

Можливі причини:

- кабель дисплея витягнутий з роз'єкта на системному блоці, або має поганий контакт з роз'єктом (поправте роз'єкт);
- несправний дисплей (відремонтуйте дисплей);
- несправний графічний контролер (замініть графічний контролер).

*Після ввімкнення живлення голосник системного блоку подає неперервний або переривчастий звуковий сигнал, операційна система не завантажується.*

Такі звукові сигнали свідчать про появу проблем у системному блоці (перевірте системну плату).

*Спроба записати файл на ГМД, або прочитати його з ГМД закінчується безуспішно*

Можливі причини:

- погана дискета (замініть дискету і повторіть спроби);
- невідформатована дискета (відформатуйте дискету);
- несправний пристрій роботи з ГМД (замініть пристрій роботи з ГМД).

*Спроба прочитати інформацію з оптичного диска закінчується безуспішно*

Можливі причини:

- поганий оптичний диск (замініть диск і повторіть спроби);
- несправний пристрій читання інформації з оптичних дисків (замініть пристрій).

***Спроба роздрукувати документ закінчується безуспішно***

Зверніть увагу на повідомлення програми, з якої ви запустили друк. Якщо ви повністю підготували пристрій друку до роботи (ввімкнули живлення, вставили папір, під'єднали кабель до пристрою друку і системного блока, пристрій друку є у списку встановлених принтерів операційної системи, а друку немає – перевірте працездатність пристрою друку й інтерфейсу приєднання до системного блока (Centronics, USB, SCSI). Для перевірки інтерфейса можна приєднати пристрій друку до іншого системного блока (від'єднання пристрою друку від інтерфейса Centronics при ввімкненому живленні може призвести до виходу з ладу інтерфейса).

***Спроба відсканувати документ закінчується безуспішно***

Зверніть увагу на повідомлення програми, з якої ви запустили сканування. Якщо ви повністю підготували сканер до роботи (ввімкнули живлення, вставили документ, приєднали кабель до сканера і системного блока, драйвер сканера встановлений на ПЕОМ), а сканування не відбувається – перевірте працездатність сканера й інтерфейса приєднання до системного блока (Centronics, USB, SCSI). Для перевірки інтерфейса можна приєднати сканер до іншого системного блока (від'єднання сканера від інтерфейса Centronics при ввімкненому живленні може призвести до виходу з ладу інтерфейса).

***Після запуску програми відтворення звуку звуковий супровід відсутній***

Можливі причини:

- ви забули ввімкнути блок живлення голосників (натисніть кнопку Power);
- кабель голосників (звукових колонок) витягнутий з роз'єднання на системному блоці, або має поганий контакт з роз'єднанням (поправте роз'єднання);
- несправні голосники (замініть їх);
- несправний звуковий канал (замініть звукову карту);
- проблеми з програмним забезпеченням звукового каналу (перевстановіть драйвер цифрового звукового каналу).



## ДОДАТОК 2. МОДЕРНІЗАЦІЯ ПЕОМ

У деяких випадках заміна несправної системної плати, модуля SIMM, ЖМД може виявитися дорожчою від придбання більш сучасної складової. Спроба замінити системну плату для I80486DX(SX) може призвести до видатків, більших від придбання системного блоку з процесором Pentium I (з частотою 66-166 МГц). Доволі проблематичною є заміна модулів SIMM. У таких випадках варто порівняти суму видатків на ремонт з вартістю модернізації ПЕОМ.

При модернізації варто врахувати такі рекомендації:

1. У корпуси з блоками живлення стандарту AT можна встановити системну плату лише такого стандарту (певний час виробники системних плат продукували системні плати з двома видами роз'ємів AT і ATX, однак такі системні плати не виробляються уже більше 5 років).

2. При виході з ладу блока живлення дешевше придбати справний користований блок живлення, аніж ремонтувати несправний. Якщо ви купуєте блок живлення не на комп'ютерній фірмі, перевірте його працездатність. Для цього візьміть із собою мережевий (для мережі 220В) шнур, ЖМД зі справними контролером і кроковим двигуном та вольтметр. Приєднайте ЖМД до блока живлення і вмикайте його. Перевірте, чи працює вентилятор блока живлення (+12В) і виміряйте вольтметром напруги +5В, +12В на роз'єдні приєднання магнітних дисків. Якщо ці напруги в нормі – купуйте блок живлення. Для перевірки блока живлення стандарту ATX потрібна системна плата цього стандарту.

3. Системні блоки іменитих фірм IBM (IBM PS/2), Compaq, Olivetti модернізуються важко, у деяких випадках модернізація просто неможлива.

4. Знайти заміну для ЖМД з інтерфейсом ST-506, ESDI (IBM PC/XT, Mazowia, Пошук-2) практично неможливо.

5. На швидкість виконання багатьох операцій істотно впливає розмір оперативної пам'яті. Здешевлення модулів оперативної пам'яті приводить до того, що з кожним роком розмір оперативної пам'яті, яку встановлюють у ПЕОМ, збільшується у кілька разів (проаналізуйте ряд 16, 32, 64, 128, 256 і 512 Мб). Якщо ваша системна плата дозволяє

встановити місткіші модулі – встановіть їх. Коли ж збільшення оперативної пам'яті потрібне, але неможливе – замініть системну плату.

6. На ЖМД бракує місця для збереження інформації – видаліть з ЖМД непотрібні вам інформацію і програми. Якщо це не допоможе, придбайте і встановіть додатковий ЖМД.

7. Промальовування графічних образів на екрані дисплея триває доволі довго – придбайте потужніший графічний контролер.

8. Ваш дисплей не підтримує режимів, які необхідні для роботи деяких програм – змініть дисплей.

9. Якість відтворення звуку двома стереоголосниками вас уже не влаштовує – придбайте звукову карту з додатковим виходами (2+1, 4+1) і відповідну акустичну систему.

10. Швидкість зчитування відеоінформації з оптичного диска недостатня для якісного відтворення відеофільму – придбайте і встановіть новий пристрій з більш високою швидкістю читання інформації.



## Видавництво «Каравела» презентує



Кравчук С. О., Шонін В. О.

### **Основи комп'ютерної техніки: Компоненти, системи, мережі**

Навчальний посібник

(гриф МОНУ №14/18.2-2544 від 01.12.04р.)

— К.: Політехніка: Каравела, 2005. — 344 с.

Формат 70x100/16

Докладно розглянуто апаратні засоби комп'ютерної техніки — від елементів комп'ютера до комп'ютерних систем та мереж. Для кожного пристрою комп'ютера описано основні принципи і технології його розроблення та функціонування, проаналізовано основні характеристики, що дозволить вибрати конкретну комплектацію комп'ютера чи мережевих засобів. Розглянуто основні компоненти розподілених інформаційних систем і способи їх з'єднання.

Для студентів вищих навчальних закладів. Буде корисним всім, хто хоче зрозуміти принципи дії апаратних засобів комп'ютерної техніки.

### • • • • • **СТИСЛИЙ ЗМІСТ** • • • • •

Розділ 1. Історія розвитку  
комп'ютерної техніки

Розділ 2. Подання даних і  
елементна база комп'ютера

Розділ 3. Компоненти комп'ютера  
3.1. Пристрої і компонування  
комп'ютера

3.2. Материнська плата

3.3. Центральний процесор

3.4. Внутрішня пам'ять

3.5. Зовнішня пам'ять

3.6. Відеосистема комп'ютера

3.7. Аудіосистема комп'ютера

3.8. Пристрої введення

3.9. Пристрої виведення

3.10. Допоміжні пристрої комп'ютера

Розділ 4. Основні класи і типи  
комп'ютерів

4.1. Суперкомп'ютери

4.2. Комп'ютери загального  
призначення і мінікомп'ютери

4.3. Мікрокомп'ютери

Розділ 5. Розподілені інформаційні  
системи

5.1. Уведення в розподілені  
інформаційні мережі

5.2. Локальні мережі

5.3. Міські та глобальні мережі



### **Інформатика. Комп'ютерна техніка. Комп'ютерні технології**

Підручник

(гриф МОНУ №1/11-1833 від 07.05.03р.)

Авт. кол.: В. А. Баженов, П. С. Венгерський,  
В. С. Гарвона, В. М. Горлач, О. М. Левченко та ін.

/ Наук. ред. Г. А. Шинкаренко, О. В. Шишов.

— К.: Каравела, 2005. — 464 с.

Формат 60x84/16



Розглянуто роботу операційних систем MS DOS, UNIX, Windows 9x. В доступній формі описано принципи роботи з програмами сімейства Microsoft Office, програмами автоматичної конвертації, перекладу та перевірки правопису тексту. Окремий розділ присвячено питанням комп'ютерної графіки. Значну увагу приділено питанням створення власних програмних продуктів за допомогою середовища візуального проектування Visual Basic 6.0. У підручнику також розглянуто основи мережевих технологій та роботу в глобальній мережі Internet.

Для студентів вищих навчальних закладів гуманітарних, соціально-економічних, інженерно-технічних, медичних, педагогічних та природничих спеціальностей, аспірантів та викладачів.

### • • • • • **СТИСЛИЙ ЗМІСТ** • • • • •

1. Основи інформатики
2. Операційні системи
3. Робота в середовищі Windows
4. Основи теорії алгоритмізації
5. Основи програмування. Visual Basic
6. Текстові процесори. Microsoft Word
7. Табличні процесори. Microsoft Excel
8. Основи роботи з базами даних. Microsoft Access
9. Комп'ютерна графіка
10. Системи створення презентацій. Microsoft PowerPoint
11. Комп'ютерні мережі. Глобальна мережа Internet
12. Перспективи розвитку інформаційних технологій

Навчальне видання

**ЗЛОБІН Григорій Григорович**  
**РИКАЛЮК Роман Євстахович**

# **Архітектура та апаратне забезпечення ПЕОМ**

Керівник видавничих проектів Ю. В. Піча  
Літературний редактор А. І. Гедзь  
Коректор В. А. Корнієнко  
Технічне редагування та комп'ютерна верстка В. С. Гарвона

Здано на складання 22.10.2005. Підписано до друку 22.03.2006.  
Формат 60×84/16. Папір офсетний. Гарнітура Тип Times. Друк офсетний.  
Ум. друк. арк. 17,79. Ум. фарбовідб. 18,54. Обл.-вид. арк. 20,2.  
Тираж 1000 прим. Зам. № 1134/091.

Видавництво «Каравела»  
просп. Рокосовського, 8а, м. Київ, 04201, Україна.  
Тел.: (044) 592-39-36. E-mail: caravela@ukr.net.  
Свідоцтво про внесення до Державного реєстру суб'єктів видавничої справи:  
ДК № 56 від 19.05.2000 р.

Якість друку відповідає наданим діапозитивам

Віддруковано у ТОВ «Навчальний друк»  
вул. Державінська, 38, м. Харків, 61001, Україна.  
Свідоцтво про внесення до Державного реєстру суб'єктів видавничої справи:  
ХК № 58 від 10.06.2002 р.

Г. Г. ЗЛОБІН, Р. Є. РИКАЛЮК

# АРХІТЕКТУРА ТА АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ ПЕОМ

У навчальному посібнику викладені історія розвитку комп'ютерної техніки, основи алгебри логіки, головна елементна база сучасних ПЕОМ. Розглянуто принципи роботи сучасних процесорів, запам'ятовуючих та периферійних пристроїв, систему шин, базову систему вводу/виводу. Подано інформацію про апаратне забезпечення ПЕОМ: відеосистему, дискову підсистему, обладнання для локальних та глобальних мереж та засоби Multimedia. Посібник призначений для студентів природничих і технічних факультетів вищих навчальних закладів та всіх, хто цікавиться інформатикою та обчислювальною технікою.

## Стислий зміст

Розділ I. Теоретичні основи функціонування ЕОМ

- 1.1. Поняття про інформацію
- 1.2. Основи алгебри логіки
- 1.3. Елементна база ЕОМ
- 1.4. Класифікація архітектур обчислювальних систем
- 1.5. Типи даних ЕОМ

Розділ II. Загальні принципи побудови ПЕОМ

- 2.1. IBM-подібні ПЕОМ
- 2.2. Особливості ПЕОМ фірми Apple
- 2.3. Особливості портативних ПЕОМ

Розділ III. Апаратне забезпечення ПЕОМ

- 3.1. Відеосистема
- 3.2. Дискова підсистема
- 3.3. Інші пристрої вводу-виводу
- 3.4. Обладнання для локальних та глобальних мереж
- 3.5. Засоби Multimedia
- 3.6. Встановлення ПЕОМ

ISBN 966-8019-41-5



9 789667 827045 >

ВИЩА ОСВІТА В УКРАЇНІ