

І. К. Сезонова, Т. П. Колісник, Ю. Є. Хорошайло

АРХІТЕКТУРА ЕОМ І МІКРОПРОЦЕСОРНІ СИСТЕМИ

Навчальний посібник

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

Харків
Оберіг
2011

УДК 004.2+004.27
ББК 32.973.2-02+32.973.26-04
С 28

Автори:

Сезонова І. К., канд. техн. наук, доцент, професор кафедри інформатики Харківського національного університету внутрішніх справ (передмова, глави 1–5, тестові завдання);

Колісник Т. П., канд. пед. наук, доцент кафедри інформатики Харківського національного університету внутрішніх справ (глава 6, тестові завдання, алфавітний покажчик);

Хорошайло Ю. Є., канд. техн. наук, доцент, заступник начальника науково-навчального інституту психології, менеджменту, соціальних та інформаційних технологій Харківського національного університету внутрішніх справ (глави 7–9)

Рецензенти:

Семенець В. В., д-р техн. наук, проф., перший проректор Харківського національного університету радіоелектроніки, лауреат Державної премії України в галузі науки і техніки;

Яковлев С. В., д-р фіз.-мат. наук, проф., завідувач кафедрою вищої математики та прикладної інформатики Харківського інституту управління;

Захаров І. П., д-р техн. наук, проф., професор кафедри метрології та вимірювальної техніки Харківського національного університету радіоелектроніки.

*Рекомендовано Міністерством освіти і науки України
як навчальний посібник для студентів вищих навчальних закладів
(лист № 1/11–1089 від 8 лютого 2011р.)*

ISBN 978-966-8689-16-1

© Сезонова І. К., Колісник Т. П., Хорошайло Ю. Є., 2011

ПЕРЕДМОВА

Електронно-обчислювальна техніка охопила сьогодні всі сфери виробництва, науки, управління та побуту. Необхідність використання сучасних досягнень у галузі мікропроцесорної техніки щодо підвищення оперативності роботи особового складу правоохоронних органів, удосконалення документообігу та управління, для проведення криміналістичних експертиз тощо не викликає сумнівів.

У системі комплексних заходів профілактики, попередження, вивчення, розкриття злочинів та посилення охорони громадського порядку важливе місце займають електронно-обчислювальні машини (ЕОМ), сучасні мікропроцесорні пристрої та системи.

Посібник з архітектури ЕОМ та мікропроцесорних систем є комплексним виданням, що одночасно знайомить читача як із загальними питаннями предметної області, так і з пристроями та системами, побудованими на основі мікропроцесорів. У посібнику розглянуто архітектури цифрових пристроїв, які набули широкого використання сьогодні та є перспективним напрямом інноваційного прогресу щодо застосування для вирішення нетипових задач завтра (наприклад, у правоохоронних органах). Посібник також розрахований на виникнення зацікавленості та ініціативи в напрямку використання ЕОМ та мікропроцесорних систем у практичній діяльності особового складу правоохоронних органів.

Практика підтвердила ефективність використання ЕОМ, комп'ютерних мереж, мікропроцесорних пристроїв і систем у діяльності правоохоронних органів. Одним із перспективних напрямків є системне, творче використання інформаційних технологій для безперервного керування та аналізу оперативної обстановки, формування управлінських рішень та контролю за їх виконанням. Іншим перспективним напрямком можна вважати впровадження нових сучасних пристроїв, побудованих на основі мікропроцесорів, для вирішення задачі збирання та опрацювання оперативних даних, безперервного та централізованого спостереження, оперативного зв'язку, захисту інформації, ідентифікації користувача ЕОМ та розкриття злочинів, пов'язаних із використанням ЕОМ та мікропроцесорної техніки.

У посібнику описано архітектуру сучасних обчислювальних машин і мікропроцесорних систем у рамках стандарту вищої освіти України. Даний навчальний посібник орієнтований на курсантів та студентів вищих навчальних закладів МВС України, які здобувають освіту за напрямом «комп'ютерні науки».

Також він буде корисним курсантам, які здобувають освіту за напрямками «системи технічного захисту інформації» і «телекомунікації», та співробітникам ОВС щодо підвищення кваліфікації.

У широкому сенсі **архітектурою** цифрової обчислювальної машини називають сукупність її властивостей і характеристик, що розглядається з погляду користувача машини.

При описі архітектури наводяться схеми, що ілюструють структуру тих або інших вузлів комп'ютера або цифрового пристрою.

У розділі I надані основні поняття предметної області, математичні основи ЕОМ та розглянуті принципи оброблення цифрової інформації.

У розділі II описані принципи побудови ЕОМ та її основних складових.

У розділі III викладена теорія мікропроцесорних систем.

Розділ IV присвячений мікроконтролерам, а розділ V – мікропроцесорам та основам їх програмування. Розділи IV та V включають необхідне теоретичне підґрунтя для вивчення системного програмування та програмування мікроконтролерів.

У розділі VI розглянуті особливості архітектури персонального комп'ютера та його периферійних пристроїв.

У розділі VII описана архітектура комп'ютерних мереж, а розділ VIII присвячений устаткуванню обчислювальних мереж, яке організує їх роботу.

В основу посібника покладений цикл лекцій з дисципліни «Архітектура ЕОМ і мікропроцесорні системи», що викладають курсантам та студентам спеціальності «інформаційні управляючі системи» автори даного посібника.

Дисципліна «Архітектура ЕОМ і мікропроцесорні системи» викладається в 4-му семестрі і є загальноосвітньою для напрямку «комп'ютерні науки». Вона розширює знання курсантів, отримані при вивченні дисциплін «Основи програмування і алгоритмічні мови», «Схемотехніка ЕОМ» та ін.

Дисципліна «Архітектура ЕОМ і мікропроцесорні системи» містить відомості про архітектуру, режими і програмне управління функціональним обладнанням і підсистемами ЕОМ; практичні заняття та вправи з численними прикладами; аналіз проблем, що виникають у галузі використання мікропроцесорних систем та технологій їх використання, і шляхів їх вирішення.

Структурно дисципліна поділяється на два змістовні модулі. Теоретичні та практичні знання курсанти отримують протягом 40 академічних годин лекцій, 16 годин практичних занять та 24 години лабораторних занять.

Цілі вивчення дисципліни:

- сформувані системне базове уявлення, уміння і навички з основ мікропроцесорних систем, достатні для подальшого продовження самоосвіти в галузі обчислювальної техніки і в суміжних галузях;

- надати відомості щодо принципів побудови, функціональних можливостей і архітектурних рішень сучасних мікропроцесорних систем, мікропроцесорів і персональних комп'ютерів.

Результатом вивчення дисципліни мають бути уміння і навички:

- роботи з вітчизняним і зарубіжним інформаційно-довідковим матеріалом;

- обґрунтування вибору апаратних засобів для вирішення конкретних прикладних задач;

- дослідження засобів і методів використання мікропроцесорних пристроїв при скоєнні протиправних діянь;

- професійного використання персонального комп'ютера і комп'ютерних мереж.

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

АЛП, ALU – арифметико-логічний пристрій
АРМ – автоматизоване робоче місце
АЦП – аналого-цифрові перетворювачі
БД – база даних
БКД – блок контролю та діагностики
БМОС – багатомашинні обчислювальні системи
БПОС – багатопроцесорні обчислювальні системи
ВВ – введення/виведення
ВІС – велика інтегральна схема
ВОЛЗ – волоконно-оптичні лінії зв'язку
ДП – динамічна пам'ять
ЕОМ – електронно-обчислювальна машина
ЕОС – електронно-обчислювальна система
ЗЗП – зовнішній запам'ятовуючий пристрій
ЗП – запам'ятовуючий пристрій
ІБ – інтерфейсний блок
ІС – інформаційна система
КК – коаксіальний кабель
КОП – код операції
КПДП – контролер прямого доступу до пам'яті
ЛК – лічильник команд
ЛОМ, LAN – локальна обчислювальна мережа
МПК – мікропроцесорний комплект
МП – мікропроцесор
МПД – мультиплексори передання даних
МПС – мікропроцесорна система
НВІС – надвелика інтегральна схема
НГМД, FDD – накопичувач на гнучкому магнітному диску
НЖМД, HDD – накопичувач на жорсткому магнітному диску
НОП – надоперативна пам'ять
ОВ – оптичне волокно
ОВС – органи внутрішніх справ
ОЗП – оперативний запам'ятовуючий пристрій
ОП, RAM – оперативна пам'ять
ОС – обчислювальна система
ПВВ – пристрої введення/виведення
ПДП – прямий доступ до пам'яті
ПЗП – постійний запам'ятовуючий пристрій
ПК – персональний комп'ютер, персональна ЕОМ
ПСт – показчик стека
ПУ – пристрій управління
РАП – регістр адреси пам'яті

РДП – реєстр даних пам'яті
РЗП – реєстри загального призначення
РК – реєстр команд
РП – реєстрова пам'ять
РС – реєстр стану
СК – симетричний кабель
СЧ – система числення
ТЗВ – технічні засоби відеоспостереження
УР – управляючі реєстри
ЦАП – цифро-аналогові перетворювачі
ЦОМ – цифрова обчислювальна машина
ЦОС – цифрова обчислювальна система
ЦП, CPU – центральний процесор
ША – шина адреси
ШДвих – вихідна шина даних
ШДвх – вхідна шина даних
ШУ – шина управління
SRAM – статична пам'ять з довільним доступом
ROM – пам'ять тільки для читання

РОЗДІЛ 1

ОСНОВИ ЕОМ

Будь-яка наукова, виробнича, культурна, суспільна діяльність пов'язана з обробленням інформації. Універсальним пристроєм для оброблення інформації, що дозволяє автоматизувати цей процес є електронно-обчислювальна машина.

Електронна обчислювальна машина (ЕОМ) – це комплекс технічних і програмних засобів, призначений для автоматизації, оброблення інформації і вирішення завдань користувачів.

1.1. ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ

Аналіз стану використання електронно-обчислювальних машин та мікропроцесорних пристроїв у правоохоронній діяльності дає можливість зробити висновок про певне відставання наявної технічної бази сучасних інформаційно-цифрових технологій від існуючих потреб. Загострення оперативної обстановки в Україні щодо збільшення кількості злочинів, вчинених із застосуванням цифрових пристроїв, збільшення об'ємів інформації, що надходить і опрацьовується, висуває на передній план питання підвищення ефективності роботи усіх служб за рахунок використання сучасних цифрових технічних засобів.

З точки зору теорії цифрових пристроїв, мікропроцесор – це найбільш складний на сьогоднішній день багатофункціональний цифровий автомат, який здатний виконувати безліч різного роду операцій.

В органах внутрішніх справ широко застосовуються програмно-апаратні засоби на базі мікропроцесорів, оскільки вони набагато полегшують, а в деяких аспектах навіть частково заміщують клопітку працю правоохоронців. Їх переважно використовують з метою відбору та оброблення інформації з різних датчиків та пристроїв у системах спостереження і захисту інформації. Актуальним є також використання специфічних цифрових пристроїв при ідентифікації, верифікації користувачів персональних комп'ютерів (ПК) та клієнтів комп'ютерних мереж.

Розкриття комп'ютерних злочинів потребує достатньо глибоких знань у галузі архітектури ЕОМ та мікропроцесорних систем. Останні, як правило, можуть виступати в ролі знаряддя злочину, що вимагає від співробітника ОВС знання їх специфічних особливостей.

Так, до найпростіших програмно-апаратних пристроїв на базі мікропроцесорів можна віднести апаратні кейлогери (пристрої, вбудовані між клавіатурою і комп'ютером, що реєструють

усі натиснення клавіш, зроблені на клавіатурі), автоматичні електронні замки з пам'яттю (які зараз використовуються не тільки в приміщеннях та автомобілях), старт-картки з вбудованими мікропроцесорами та ін.

До більш складних систем, без яких у сучасних умовах дуже важко уявити нормальну діяльність працівників ОВС, потрібно зарахувати: дактилосканери (для зняття і сканування відбитків пальців), цифрові засоби зняття інформації (цифрові фотоапарати, відеокамери, пристрої для вимірювання швидкості та ідентифікації транспортного засобу), біометричні паспорти, охоронні та пожежні сигналізації високих рівнів захисту та багато інших.

Управління такими пристроями або системами пристроїв, а в майбутньому – розширення їх функціональних можливостей значною мірою залежить від рівня підготовки спеціалістів у даній галузі знань.

1.2. ОСНОВНІ ПОНЯТТЯ ТЕОРІЇ ЕОМ

Архітектурою ЕОМ прийнято вважати сукупність принципів системної, функціональної, логічної і фізичної організації апаратних і програмних засобів ЕОМ.

Ускладнення периферії ЕОМ і поява багатопроцесорних систем породили терміни «обчислювальні системи», «платформи», «обчислювальні середовища» тощо, інтуїтивне ж поняття ЕОМ як найповніше – це цифрова обчислювальна система, виконана як єдиний конструктив.

З іншого боку, у поняття ЕОМ можна включити і спеціалізований мікропроцесор управління «рукою» робота, і персональний комп'ютер, і суперкомп'ютер, який як елемент мережі може іменуватися також майнфреймом. Загальним є функціональне призначення ЕОМ – забезпечення потреб прикладної системної області. При цьому використання ЕОМ розглядається в основному як універсальні обчислювальні системи (платформи).

Наведемо основні поняття, які використовуються в теорії ЕОМ.

Система (від грец. слова «ціле») – це сукупність взаємодіючих елементів, які діють разом та утворюють певну єдність. Існує ряд інших визначень, з яких одне найвідоміше: система складається з підсистем, які мають властивості системи.

Елементом системи називається така її частина, яка має певне функціональне призначення. Якщо складовий елемент системи у свою чергу складається з простих взаємозв'язаних елементів, то такий складний елемент системи називають підсистемою.

Організація системи – це внутрішня впорядкованість, узгодженість взаємодії елементів системи, які діють разом. Взаємодія елементів системи визначає можливі стани системи. Кількість таких станів обмежена.

Структура системи – це склад, порядок і принципи взаємодії елементів системи, що визначають основні властивості системи. У структурі системи відокремлюють сукупності елементів, які функціонують разом (різні рівні ієрархії) і описують функціонування елементів одного рівня ієрархії незалежно від іншого.

Цілісність системи – необхідна властивість системи, яка забезпечує її адекватну роботу.

Властивості системи не дорівнюють сумі властивостей її елементів (у системі виявляється додатковий ефект взаємодії підсистем). В той же час властивості елементів системи залежать від місця елемента і його функцій усередині системи.

Архітектура системи – сукупність властивостей системи, істотних для користувача.

Архітектура ЕОМ – абстрактне представлення комп'ютера, яке відображає його структуру, схемотехніку і логічну організацію.

Архітектура ЕОМ – це багаторівнева ієрархія апаратно-програмних засобів, з яких будується ЕОМ. Кожен з рівнів допускає багатоваріантну побудову і застосування. Конкретна реалізація рівнів визначає особливості структурної побудови ЕОМ.

Структура – сукупність елементів і їх зв'язки. Розрізняють структури технічних, програмних і апаратно-програмних засобів.

Мікропроцесорна система (МПС) – сукупність апаратних і програмних засобів, побудованих на основі мікропроцесора (мікроконтролера) для вирішення конкретної задачі. Мікропроцесорна система (МПС) – управляюча інформаційна або інша спеціалізована цифрова система, побудована на базі мікропроцесора.

Визначимо поняття та терміни, які використовуються в мікропроцесорній техніці:

Мікропроцесор (МП) – програмно керований пристрій, що здійснює процес оброблення цифрової інформації і управління ним; побудований на одній або кількох великих інтегральних схемах (ВІС).

Мікропроцесорна ВІС – інтегральна мікросхема, що виконує функцію МП або його частини (ВІС з процесорною організацією, розроблена для побудови мікропроцесорних систем).

Цифровий пристрій – це МПС, що складається з мікропроцесора (мікропроцесорів), пам'яті, засобів зв'язку з периферійними

пристроями та джерела живлення, поєднаних загальною несучою конструкцією.

Мікропроцесорний комплект (МПК) – набір мікропроцесорних та інших інтегральних мікросхем, сумісних за конструктивно-технологічним виконанням та призначених для сумісного використання при побудові цифрових пристроїв.

1.3. ЕОМ ЯК ПРИСТРІЙ ДЛЯ ОБРОБЛЕННЯ ЦИФРОВОЇ ІНФОРМАЦІЇ

Інформаційною системою (ІС) називають систему, яка організує накопичення, зберігання і перетворення інформації в залежності від функцій, які вона виконує. З іншої точки зору, під інформаційною системою розуміють комплекс взаємозалежних методів і засобів перетворення даних, які необхідні користувачеві ІС (структурне визначення ІС).

Інформація – це набір відомостей про явища, події, процеси, стани, кількість об'єктів навколишнього середовища. Основна частина цієї інформації зберігається в будь-якій формі в пам'яті до моменту потреби в її використанні. У системах автоматизованого оброблення інформації її передання відбувається за допомогою сигналів.

Сигнал – це фізичний процес, що може бути аналоговим (який безперервно змінюється за амплітудою і в часі) або дискретним (цифровим) (який може приймати кінцеве число значень у певні моменти часу). Аналогові сигнали використовуються в телефонах, радіомовленні, телебаченні. Сигнали, які несуть текстову, символну інформацію, дискретні. Різні компоненти ЕОМ є цифровою (дискретною) технікою і можуть обробляти інформацію, представлену тільки в чисельній формі. При введенні кожна буква кодується певним числом. Процес встановлення відповідності між інформацією, що вводиться, і числами називається кодуванням символів. Одиницею інформації є один біт, тобто двійковий розряд, що може приймати значення 0 й 1. Вісім біт становлять один байт. В одному байті можна закодувати значення одного символу з 256 можливих ($256 = 2^8$).

Залежно від рівня механізації процедур оброблення інформації розрізняють системи ручного оброблення, механізованого (сортування аркушів), автоматизованого і автоматичного (автоматизовані інформаційні системи). Надалі будемо під ІС розуміти саме автоматизовані інформаційні системи. Основу будь-якої автоматизованої ІС складає ЕОМ (вітчизняна термінологія) або комп'ютер (закордонна термінологія). ЕОМ – це комплекс технічних засобів, призначених для автоматичного оброблення інформації в процесі вирішення обчислювальних та інформаційних задач.

Перші ЕОМ були створені наприкінці 40-х рр. 20 ст. і були за масштабами гігантськими монстрами, які використовувалися тільки для обчислювальних цілей. Елементною базою цих комп'ютерів були електронні лампи, у зв'язку з чим вони не тільки займали великі площі, але й виділяли велику кількість тепла. У процесі вдосконалення технологій ЕОМ зменшилися в розмірах і збільшили кількість додаткового устаткування, що розширило сферу їх застосування. У 70-х рр. 20 ст. ЕОМ перетворилися спочатку на обчислювальні системи, а потім на інформаційно-обчислювальні системи.

Обчислювальна система – це сукупність одного або декількох комп'ютерів чи процесорів, програмного забезпечення і периферійного устаткування яке налаштоване на виконання інформаційно-обчислювальних задач.

Перші обчислювальні системи створювалися з метою збільшення швидкості дії і надійності роботи шляхом паралельного виконання обчислювальних операцій.

Еволюцію обчислювальної техніки з 50-х рр. 20 ст. по теперішній час можна простежити щодо сфер використання, режиму роботи, професійної підготовленості користувача та інтерфейсу (див. табл. 1.1)

Таблиця 1.1

Еволюція ЕОМ

Період	Характер використання	Режим роботи	Тип користувача	Тип діалогу
1	2	3	4	5
50-ті рр. XX ст.	Науково-технічні розрахунки	Одно-програмний	Інженер-програміст	Робота за пультом
60-ті рр. XX ст.	Технічні і економічні розрахунки	Пакетне оброблення	Інженер-програміст	Перфокарти, перфострічки і роздруковані машинограмами
70-ті рр. XX ст.	Управління і економічні розрахунки	Розподіл часу	Програмісти	Інтерактивний (через клавіатуру і екран)
80-ті рр. XX ст.	Управління, надання інформації	Персональна робота	Користувачі з технічною освітою	Інтерактивний за жорстким меню
90-ті рр. XX ст.	Телекомунікації, інформаційне обслуговування і управління	Мережне оброблення	Без обмежень	Інтерактивний з інтелектуальним меню

Закінчення табл. 1.1

1	2	3	4	5
---	---	---	---	---

початок XXI ст.	Усі сфери професійної діяльності, розваг, послуги Інтернету	Мережне оброблення	Без обме- жень	Інтерактивний з інтелектуальним меню
--------------------	--	-----------------------	-------------------	--

1.4. ФОРМИ ПРЕДСТАВЛЕННЯ ІНФОРМАЦІЇ В ЕОМ

Відомо, що в ЕОМ інформація відображається за допомогою бітів, які приймають значення 0 або 1. Говорять, що ЕОМ оперує двійковою системою числення.

У математиці відомі позиційна і непозиційна системи числення.

Двійкове представлення числа вимагає приблизно в 3,3 рази більшого числа розрядів, ніж його десяткове уявлення. Проте двійкове кодування легко реалізується технічно, оскільки електронні схеми для обробки двійкових кодів повинні знаходитися тільки в одному з двох станів (є сигнал/немає сигналу або висока напруга/низька напруга). Програмістів двійкове кодування привертає тим, що дозволяє представити інформацію будь-якої природи (тексти, картинки, фільми, звукозаписи) в одному і тому ж двійковому вигляді, придатному для оброблення на ЕОМ.

Система числення (СЧ) – це спосіб найменування і зображення чисел за допомогою символів, що мають певне кількісне значення.

У непозиційній системі числення цифри не змінюють свого кількісного значення залежно від позиції в зображенні числа. Так, непозиційне римське X має значення «десять», у числі XXX – «тридцять», і в числі LX – «шістдесят». Непозиційні системи числення не набули широкого поширення через складні правила визначення значення за зображенням та громіздкість самого зображення.

У позиційній СЧ з основою p кількісне значення кожної цифри залежить від її місця (позиції) в числі. Кількість $p - 1$ різних цифр, які використовуються для зображення числа в позиційній СЧ, називається основою СЧ. Значення цифр лежать у діапазоні від 0 до $(p - 1)$.

У загальному випадку запис будь-якого числа в СЧ з основою p представляється у вигляді ряду:

$$N = a_{m-1} \times p^{v-1} + a_{m-2} \times p^{m-2} + \dots + a_1 \times p + a_0 + a_{-1} \times p^{-1} + a_{-2} \times p^{-2} + \dots + a_{-s} \times p^{-s}, \quad (1)$$

де нижні індекси визначають місцеположення цифри в числі (розряд):

– позитивні значення індексів, включаючи нуль – для цілої частини числа (m розрядів);

– негативні значення – для дробної частини числа (s розрядів).

Максимальне ціле число, яке може бути представлено в m розрядах:

$$N_{\max_{ц}} = p^m - 1. \quad (2)$$

Мінімальне значення (не 0), яке можна записати в s розрядах дробної частини:

$$N_{\min_{др}} = p^s + 1. \quad (3)$$

У загальному випадку, маючи p розрядів цілої частини числа і s – дробної, можна записати p^{m+s} різних чисел. Саме цей вираз представляє діапазон значень, якими оперує позиційна СЧ.

ЕОМ оперує з числами, що містять кінцеву кількість розрядів. Розрядність ЕОМ і визначає обмеження щодо довжини інформаційного слова, яке обробляє машина одночасно. У теорії ЕОМ використовується термін «довжина розрядної сітки машини», під якою розуміється сукупність бітів інформації, призначених для зберігання одного машинного слова. Тобто діапазон представлення чисел в ЕОМ обмежений довжиною машинних слів.

1.4.1. Двійкова система числення

Двійкова система числення має основу $p = 2$ і використовує для представлення інформації дві цифри – 0 і 1. Існують правила переведення чисел з однієї системи числення в іншу, засновані на співвідношенні (1).

Наприклад, двійкове число 101110,101 дорівнює десятковому 46,625.

У співвідношення (1) замість a_m підставляються відповідні розряди числа 101110,101 та підраховуються коефіцієнти, які представляють собою вагу кожного двійкового розряду та дорівнюють відповідному степеню числа 2 (основи СЧ).

Отримуємо:

$$101110,101_2 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} = 32 + 8 + 4 + 2 + 0,5 + 0,125 = 46,625_{10}.$$

Сформулюємо правило переведення числа з двійкової СЧ у десяткову.

Для переведення числа з двійкової системи числення в десяткову необхідно:

- 1) визначити вагу кожного двійкового розряду в десятковій СЧ;
- 2) підрахувати суму, яку отримаємо, склавши значення ваги при одиничних значеннях розрядів двійкового числа (не забувайте, що абсолютні значення ваг збільшуються справа наліво в цілій частині і зліва направо – у дробовій).

Алгоритм переведення числа з десяткової СЧ у систему числення з основою p дозволяє оперувати з числами в десятковій СЧ (тій СЧ, з якої число переводиться) і передбачає окреме переведення цілої і дробної частин:

– для переведення цілої частини числа початкове число, а потім і цілі частини, які з'являються в процесі перетворень, слід послідовно ділити на основу p до тих пір, поки ціла частина частки не буде дорівнювати 0. Залишки від ділення, записані послідовно справа наліво (від останніх до перших), утворюють цілу частину числа в СЧ з основою p ;

– для переведення дробної частини початкове число, а потім дробні частини, які з'являються в процесі перетворень, слід послідовно множити на основу p до тих пір, поки чергова дробна частина числа не буде дорівнювати 0 (число переводиться в нову СЧ точно) або не буде досягнута задана точність (якщо після s кроків дробна частина не дорівнює 0, число переводиться в нову СЧ неточно). Цілі числа, записані після коми послідовно зліва направо (у порядку отримання, від перших до останніх), утворюють дробну частину числа в СЧ з основою p .

Приклад. Згідно з вищевикладеними правилами, переведення у двійкову систему числа 46,625 буде наступним:

1. Ціла частина числа дорівнює 46.

Прослідкуємо процес ділення:

Крок 1. $46 : 2 = 23$ (залишок 0)

Крок 2. $23 : 2 = 11$ (залишок 1)

Крок 3. $11 : 2 = 5$ (залишок 1)

Крок 4. $5 : 2 = 2$ (залишок 1)

Крок 5. $2 : 2 = 1$ (залишок 0)

Крок 6. $1 : 2 = 0$ (залишок 1).

2. Записуємо залишки в зворотному порядку: 101110.

3. Дробна частина числа дорівнює 0,625.

Прослідкуємо процес множення:

Крок 1. $0,625 \times 2 = 1,250$ (ціла частина 1)

Крок 2. $0,25 \times 2 = 0,5$ (ціла частина 0)

Крок 3. $0,5 \times 2 = 1,0$ (ціла частина 1)

4. Записуємо цілі частини в порядку отримання: 101.

У теорії ЕОМ також використовують шістнадцятирічну систему числення, яка є найбільш зручною для завдання адрес комірок пам'яті.

Переведення з десяткової системи числення в шістнадцятирічну, крім наведеного вище правила, можна здійснити в два етапи: спочатку перевести число у двійкову систему числення, а потім кожні чотири двійкових розряди записати шістнадцятковими цифрами.

Так, за допомогою чотирьох двійкових чисел представимо шістнадцятирічні:

0000 – 0	0001 – 1	0010 – 2	0011 – 3
0100 – 4	0101 – 5	0110 – 6	0111 – 7
1000 – 8	1001 – 9	1010 – A	1011 – B
1100 – C	1101 – D	1110 – E	1111 – F

Щоб кількість розрядів була кратна 4, можна додавати незначущі нулі зліва до цілої частини числа і справа – до дробної. Використовуючи викладені вище за положення, можна записати рівняння:

$$00101110,1010_2 = 2E,A_{16},$$

де нижні індекси вказують на СЧ, у якій записане число.

1.4.2. Представлення чисел з фіксованою і рухомою крапкою

Для представлення чисел в ЕОМ використовуються дві форми: натуральна (з фіксованою крапкою) і нормальна або напівлогарифмічна (з рухомою крапкою).

Натуральна форма припускає, що положення коми, яка відокремлює цілу частину числа від його дробної частини, є фіксованим у розрядній сітці ЕОМ. Крім того, ці числа можуть бути беззнаковими (позитивними) або мати знак, для представлення якого виділяється спеціальний розряд – знаковий. Як правило, це найстарший розряд у числі (крайній лівий). Для додатніх чисел у знаковий розряд заноситься 0, для від’ємних – 1.

У ЕОМ відомо два способи фіксації положення крапки – або перед старшим розрядом числа (після знакового розряду, якщо він є), або після молодшого розряду. Розрядна сітка ЕОМ для цих варіантів може бути представлена так, як показано на наведених нижче ілюстраціях (див. рис. 1.1–1.4). У прямокутниках, що показують розряди двійкових чисел, проставлені ваги розрядів.

2^{-1}	2^{-2}	...	2^{-n}
----------	----------	-----	----------

Рис. 1.1. Число без знаку з крапкою, фіксованою перед старшим розрядом

Знак	2^{-1}	2^{-2}	...	$2^{-(n-1)}$
------	----------	----------	-----	--------------

Рис. 1.2. Число із знаком та крапкою, фіксованою перед старшим розрядом

2^n	2^{n-1}	...	2	2^0
-------	-----------	-----	---	-------

Рис. 1.3. Число без знаку з крапкою, фіксованою після молодшого розряду

Знак	2^{n-1}	2^{n-2}	...	2	2^0
------	-----------	-----------	-----	---	-------

Рис. 1.4. Число зі знаком та крапкою, фіксованою після молодшого розряду

У першому варіанті можуть бути представлені тільки правильні дроби, у другому – цілі числа. Змішане число в такій розрядній сітці відобразити не можливо.

Нормальна форма представлення чисел дозволяє відображати змішані числа і, крім того, значно розширює діапазон представлених чисел. Ця форма ґрунтується на наступному уявленні числа:

$$X = A^p \times M, \quad (4)$$

де X – напівлогарифмічна форма уявлення числа, p – порядок, A – основа порядку, M – мантиса, яка визначає числове уявлення.

Числа представляються у вигляді мантиси і степеня (порядку) числа у двійковій системі числення. Мантиса зазвичай представляється правильним дробом у нормалізованому вигляді (звідси і назва форми уявлення). У нормалізованому вигляді перша цифра мантиси (праворуч від коми) повинна бути відмінною від нуля. Це дозволяє уникнути множинних форм представлення чисел з рухомою крапкою. Розрядна сітка ЕОМ для такого уявлення може бути наступною:

Знак числа	2^{-1}	2^{-2}	...	2^{-m}	Знак порядку	2^p	2^{p-1}	...	2^0
---------------	----------	----------	-----	----------	-----------------	-------	-----------	-----	-------

Рис. 1.5. Розрядна сітка ЕОМ для представлення чисел у вигляді мантиси і порядку

При $m = 22$ і $p = 10$ діапазон чисел складає від 10^{-300} до 10^{+300} .

Для порівняння: кількість секунд, які пройшли з моменту утворення планети Земля, складає всього 10^{18} .

Зазначемо, що всі числа із рухомою крапкою представлені в ЕОМ у нормалізованому вигляді. Нормалізованими називають числа, у яких старший розряд не дорівнює 0. У нормалізованих двійкових чисел мантиса M вкладається в діапазон $0,5 \leq M < 1$.

Питання для самоперевірки:

1. Наведіть приклади використання цифрових пристроїв.
2. Які ви знаєте пристрої для реєстрації інформації?
3. Наведіть приклади цифрових пристроїв для ідентифікації особи.
4. Дайте визначення системи та елементу системи.
5. Що таке архітектура ЕОМ?
6. Які системи числення використовуються в ЕОМ для представлення інформації?
7. Наведіть порозрядні схеми представлення чисел із комою в ЕОМ.

РОЗДІЛ 2

ПРИНЦИПИ ПОБУДОВИ ЕЛЕКТРОННО-ОБЧИСЛЮВАЛЬНИХ МАШИН

2.1. Принцип дії ЕОМ

Обчислювальна система є перш за все пристроєм для оброблення інформації.

ЕОМ може обробляти інформацію, представлену тільки в чисельній формі. Будь-яка інформація (текстова, числова) кодується певним числом. Процес встановлення відповідності між інформацією і числами, за допомогою яких вона відображається в пам'яті ЕОМ, називається **кодуванням**.

В основу роботи ЕОМ покладена її спроможність виконувати послідовність команд. Послідовність команд, у свою чергу, утворює програму.

ЕОМ можуть мати різні конструктивні особливості, відрізнятись продуктивністю, точністю, кількістю пристроїв кожного типу, способом введення/виведення інформації, але всі вони у своєму складі мають не менше трьох основних функціональних пристроїв: процесорний пристрій (процесор); зовнішню пам'ять великого об'єму; канал обміну інформацією із зовнішньою пам'яттю і пристроями введення/виведення.

Принцип програмного управління

Сучасні ЕОМ будуються на одному принципі – принципі програмного управління. Цей принцип ґрунтується на можливості представлення алгоритму у формі команд, які задають спосіб оброблення машинних слів і мають два типи: операторів, що забезпечують перетворення інформації, і операторів, що аналізують інформацію з метою визначення порядку виконання програми. Реалізація цього принципу в різних ЕОМ може бути різною. Принцип програмного управління, який використовується в сучасних ЕОМ, був запропонований у 1945 р. американським математиком Дж. фон Нейманом, і з тих пір нейманівський принцип програмного управління використовується як основний принцип побудови ЕОМ.

Цей принцип включає наступні положення.

1. Інформація кодується у двійковій формі і розділяється на одиниці (елементи) інформації, які ми називаємо **словами**.

2. Різнотипні слова (числа, символи, команди) розрізняються за способом використання, а не способами кодування.

3. Слова розміщуються в комірках пам'яті машини і ідентифікуються номерами цих комірок, які ми називаємо **адресами слів**.

4. Алгоритм представляється у формі послідовності слів (команд), які визначають тип операції і слова інформації, що беруть участь в операції, і називаються **командами**. Алгоритм, представлений у термінах машинних команд, називається **програмою**.

5. Команди виконуються послідовно, у відповідності з їх розташуванням у пам'яті.

Використання в ЕОМ двійкових кодів обумовлене специфікою електронних схем, які забезпечують передання, зберігання і перетворення інформації. Сукупності бітів інформації, які використовуються для уявлення окремих чисел, команд і т. ін., розглядаються як самостійні інформаційні об'єкти і називаються словами. Слово обробляється в ЕОМ як одне ціле – як машинний елемент інформації.

Згідно з другим положенням, усі слова в ЕОМ виглядають абсолютно однаково. Відрізняються вони тільки способом використання. Завдяки такій зовнішній одноманітності слів стає можливим використовувати одні й ті ж операції для оброблення слів різної природи, наприклад, і чисел, і команд.

Третє положення фіксує специфіку зберігання та ідентифікації інформації, яка ґрунтується на конструктивних особливостях машинної пам'яті. Машинна пам'ять – це сукупність комірок, кожна з яких є місцем для зберігання слова інформації, як синонім цього терміна можна запропонувати «сховище інформації». Комірка пам'яті забезпечує зберігання окремого слова, зокрема константи або команди. Щоб записати слово в пам'ять, необхідно вказати адресу комірки, яка відведена для його зберігання. Щоб вибрати слово з пам'яті (прочитати його), слід знову ж таки вказати адресу комірки пам'яті. Таким чином, адреса комірки стає машинним ідентифікатором (ім'ям) слова даних. Для ідентифікації даних і команд в ЕОМ немає ніяких інших засобів, окрім адрес, які привласнюються даним і командам у процесі написання програми обчислень.

Крім того, відзначимо, що вибірка (читання) слова з пам'яті не руйнує інформацію, що зберігається в комірці. Це дозволяє будь-яке слово, що записане, читати скільки завгодно разів, тобто з пам'яті вибираються не слова, а копії слів. Також слід наголосити, що адреса комірки може бути представлена в ЕОМ так само як і будь-яке інше слово інформації, яке використовується в операціях.

У четвертому положенні принципу програмного управління стверджується, що програма представляється в ЕОМ у вигляді впорядкованої лінійної послідовності команд наступного вигляду:

1 ... r	1 ... m	1 ... m		1 ... m
КОП	A ₁	A ₂	...	A _k

Рис. 2.1. Формат команди

Представлена послідовність дає уявлення, як виглядає машинне слово, частини якого називають **полями**. Так, КОП, $A_1, A_2 \dots A_k$ – це поля команди, що представляють відповідно код операції (КОП) і адреси операндів $A_1, A_2 \dots A_k$, що беруть участь в операції. Над кожною коміркою таблиці вказані номери двійкових розрядів полів: код операції складається з r бітів, кожне поле адреси операндів $A_1, A_2 \dots A_k$ містить m бітів. Наведене зображення команди характеризує **структуру** або, інакше кажучи, **формат** команди.

Згідно з п'ятим положенням, пам'ять нейманівської машини суто лінійна. Це відтворюється в процесі обчислень, які виконує ЕОМ за програмою. Виконання програми починається з команди, яка задана пусковою (початковою) адресою програми. За нею виконується наступна в пам'яті команда. Послідовність порушується тільки в тому випадку, якщо виконується команда умовного або безумовного переходу, у якій безпосередньо вказується адреса наступної команди. Процес обчислень продовжується до тих пір, доки не буде виконана команда HLT*, яка припиняє обчислення.

Нейманівський принцип програмного управління не є єдино можливим принципом функціональної організації ЕОМ. Учені неодноразово декларували необхідність перегляду нейманівського принципу, оскільки можливості таких машин доведені до межі. Проте і в паралельних, і конвеєрних ЕОМ процесори працюють за принципом Дж. фон Неймана, що є переконливим підтвердженням того, що цей принцип є найбільш раціональним принципом побудови ЕОМ.

Викладені принципи свідчать про те, що ЕОМ відноситься до класу систем програмного управління, а безпосередній ініціатор виконання чергової операції – це команда.

Абсолютна більшість ЕОМ відповідають п'яти принципам фон Неймана і відносяться сьогодні до класу машин, якими управляють потоки команд. Програма в таких машинах починає виконуватись за необхідністю вирішення задачі, тобто за запитом. Спочатку із пам'яті зчитується команда, а потім за наведеними адресами визначаються дані (операнди).

Останнім часом з'явився новий клас ЕОМ, якими управляють потоки даних. Кожна команда програми не буде виконуватись

* Команда HLT (від англ. halt – зупинка, рос. – «останова») – це команда призупинки процесора доти, доки не виникне апаратне переривання. Коли процесор знаходиться на стадії виконання цієї команди, він споживає мінімальну кількість енергії, а отже, майже не виділяє тепла.

Команду HLT не слід плутати із командою NOP, або NOOP (скорочення від англ. no operation), яка наказує нічого не робити.

до тих пір, поки не активізуються операнди (дані). Таким чином, спочатку повинні бути прочитані всі операнди, а потім готова до виконання команда буде виконуватись. Очевидно, що робота ЕОМ, якими управляють потоки даних, не відповідає принципам фон Неймана, за винятком першого.

Програми, записані в пам'ять, складають програмне забезпечення ЕОМ і є її необхідною частиною. Для виконання команд використовуються як апаратні, так і програмні засоби. Межі між цими засобами постійно змінюються. Новітні технології дозволяють удосконалювати апаратну частину. Розвиток програмних засобів не потребує додаткових апаратних витрат, за винятком додаткової пам'яті для нових програм.

Система команд звичайно не охоплює всі можливі апаратні дії, тому в обчислювальних пристроях широко використовується мікропрограмне управління, яке займає проміжне місце між програмним і апаратним управлінням. Мікропрограмне управління – це послідовне виконання мікрокоманд, які зберігаються в спеціальній постійній пам'яті. Кожна мікрокоманда – це набір управляючих сигналів. Послідовність виконання мікрокоманд програмується шляхом запису в мікрокомандах адреси наступних мікрокоманд. Мова мікропрограмного оброблення – виключно спеціальна; вона може бути орієнтована на будь-яку частину апаратури. Мікропрограмне оброблення стандартизує і скорочує об'єм апаратури, тому що збільшення функцій обчислювальної системи (ОС) здійснюється за рахунок удосконалення мікропрограмного забезпечення. Швидкість мікропрограмного оброблення вища за швидкість програмного оброблення завдяки його спеціалізації і локалізації, але, звичайно, нижча за апаратне.

Для кращого розуміння організаційних принципів роботи ОС доцільно ввести поняття процесу. **Процес** – це виконання програми. Для забезпечення процесу необхідна наявність засобів: оперативної пам'яті, у яку записані програма, вхідні дані і результати обчислень; апаратура для виконання команд програми з мікропрограмним або будь-яким іншим видом управління; необхідний набір управляючих програм. Необхідною є також процедура запису в пам'ять програми та вхідних даних; перетворення програми на форму, яку розуміє апаратура (машинною мовою); завдання режиму роботи ОС.

У кожний момент часу процес знаходиться у визначеному стані та в будь-який момент часу може бути зупинений. Для його відновлення необхідні апаратні і програмні засоби для зберігання та аналізу інформації, яка характеризує стан процесу.

ОС в змозі одночасно реалізовувати з перериваннями декілька процесів. У такому випадку потрібно організувати певну взаємодію процесів. Програми, які беруть участь у процесі, можуть використовувати одні й ті ж області пам'яті та інші ресурси. Тобто для реалізації своїх функцій обчислювальній системі необхідно мати комплекс програмних засобів, який називають **математичним (програмним) забезпеченням**.

Математичне забезпечення комп'ютера, наприклад, складається з операційної системи, програм технічного обслуговування і пакетів прикладних програм.

Операційна система – це основна частина математичного забезпечення ЕОМ, призначена для управління обчислювальним процесом разом з апаратурою. Основні функції операційної системи: розподіл ресурсів ЕОМ, управління режимами роботи, запуск програм на виконання, програмна підтримка діалогового режиму роботи і т. ін. Перед виконанням програми проходять процедуру трансляції, тобто переклад з мов високого рівня на мову машинного рівня.

Програми технічного обслуговування автоматизують перевірку працездатності всіх пристроїв ЕОМ та виконання програм. До складу програм технічного обслуговування входять, наприклад, програми для діагностики апаратної частини ЕОМ.

Пакети прикладних програм – це програмні засоби для вирішення широкого класу задач, наприклад, науково-технічних або економічних. Склад і об'єм прикладних програм постійно розширюється і змінюється. До прикладних програм також відносяться програми, які розширюють функції операційних систем.

Сукупність апаратних і програмних засобів, організованих за вищевикладеними правилами, і утворюють обчислювальну систему.

2.2. ХАРАКТЕРИСТИКИ ЕОМ

Однією з найважливіших характеристик ЕОМ є її *швидкодія*, яка характеризується кількістю команд, які виконує ЕОМ за одну секунду. Оскільки до складу команд ЕОМ включаються операції, різні за тривалістю виконання і за вірогідністю їх використання, то має сенс характеризувати її або середньою швидкодією ЕОМ, або граничною (для «найкоротших» операцій типу «регістр-регістр»). Сучасні обчислювальні машини мають дуже високі характеристики за швидкодією, які вимірюються сотнями мільйонів операцій за секунду. Наприклад, новітній мікропроцесор Merced спільного виробництва фірм Intel і

Hewlett-Packard, має пікову продуктивність більше мільярда операцій за секунду.

Іншою найважливішою характеристикою ЕОМ є *ємкість за-пам'ятовуючих пристроїв*. Цей показник дозволяє визначити, який об'єм програм і даних може бути одночасно розміщений у пам'яті. У даний час персональні ЕОМ можуть мати ємкість оперативної пам'яті в розмірі декількох гігабайт. Цей показник дуже важливий для визначення, які програмні пакети можуть одночасно оброблятися в машині.

Надійність – це здатність ЕОМ за певних умов виконувати необхідні функції протягом заданого періоду часу. Наприклад, у сучасних вінчестерів HDD середній час напрацювання на відмову досягає 500 тис. год (близько 60 років).

Точність – можливість розрізняти майже рівні значення. Точність отримання результатів оброблення інформації в основному визначається розрядністю ЕОМ, а також структурними одиницями представлення інформації (байтом, словом, подвійним словом). За допомогою засобів програмування (мов високого рівня) цей діапазон може бути збільшений у декілька разів, що дозволяє досягати дуже високої точності.

Достовірність – властивість інформації бути правильно сприйнятою. Достовірність характеризується вірогідністю отримання безпомилкових результатів. Заданий рівень достовірності забезпечується апаратно-програмними засобами контролю самої ЕОМ. Існують методи контролю достовірності шляхом вирішення еталонних задач і повторних розрахунків. В особливо відповідальних випадках проводяться контрольні вирішення на інших ЕОМ і порівняння результатів.

2.3. Класична схема ЕОМ

Викладені принципи програмного управління (фон Неймана) реалізовані в апаратному забезпеченні ЕОМ, структура якого поступово оформилася в наведену нижче схему (див. рис. 2.2), що стала сьогодні класичною і складається з наступних блоків:

- блок для виконання логічних і арифметичних операцій (АЛП);
- блок для зберігання інформації (пам'ять або ОЗП);
- пристрої для введення і виведення даних;
- пристрій управління для забезпечення узгодженої роботи вище перелічених пристроїв перетворення інформації (ПУ).

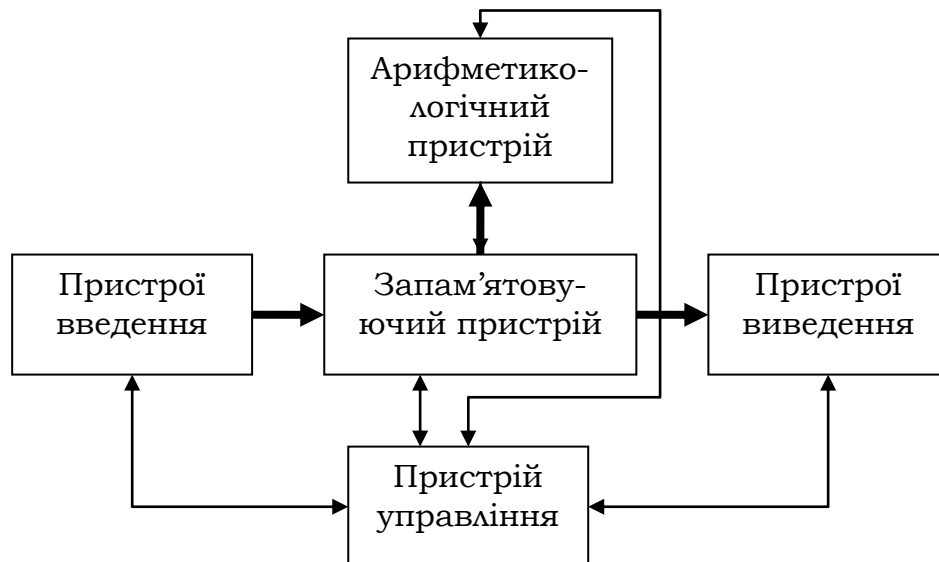


Рис. 2.2. Класична схема ЕОМ

Арифметико-логічний пристрій (АЛП) виконує арифметичні і логічні операції з даними. Пристрій управління (ПУ) організує взаємодію всіх складових ЕОМ. Запам'ятовуючий пристрій (ЗП), або пам'ять, слугує для зберігання даних і програм, а також для тимчасового зберігання даних і результатів операцій. Пристрої введення і виведення інформації забезпечують зв'язок із навколишнім середовищем, забезпечують передання вхідної інформації від користувача до ЕОМ та виведення результатів обчислень у зовнішнє середовище.

На схемі лініями різної товщини позначені потоки інформації (слів даних) та управляючих сигналів. Потоки інформації позначені лініями більшої товщини.

Пристрій управління ініціює роботу пристроїв введення, даючи команду введення в пам'ять. Аналогічно ініціюється робота пристроїв виведення.

Пристрій управління вказує, з якого елемента пам'яті ЗП необхідно передавати інформацію в АЛП, яку операцію над цією інформацією повинен виконати АЛП, у яку комірку пам'яті потрібно занести результат операції.

2.4. МАГІСТРАЛЬНО-МОДУЛЬНИЙ ПРИНЦИП ОРГАНІЗАЦІЇ ЕОМ

Сучасні ЕОМ мають відмінності обумовлені розвитком обчислювальної техніки. Запам'ятовуючий пристрій представляється кількома рівнями: внутрішній, або оперативний, ЗП (ОЗП) і зовнішній ЗП (ЗЗП). Внутрішній ЗП містить інформацію, що обробляється за визначений проміжок часу, включаючи і поточний момент. Зовнішній ЗП служить сховищем усієї інформації конкретного користувача.

АЛП і ПУ об'єднані в один пристрій – центральний процесор (ЦП, або CPU).

Центральним процесором (ЦП) називають пристрій, що безпосередньо здійснює процес оброблення даних і програмне управління цим **процесом**. До його складу входять арифметико-логічний пристрій (АЛП), пристрій управління (ПУ) і власна (реєстрова) пам'ять процесора. У сучасних ПК центральний процесор ЦП реалізований у вигляді великої інтегральної схеми і називається мікропроцесором.

Центральний процесор взаємодіє з ОЗП або просто оперативною пам'яттю (ОП). Оперативна пам'ять призначена для прийому, зберігання і видання інформації (чисел, символів, команд, констант) – усієї інформації для виконання обчислень за програмою.

У сучасних персональних комп'ютерів є досить великий арсенал пристроїв введення і пристроїв виведення (ПВВ). Структурно сучасні ПК складається з 2-х частин: центральної і периферійної. Головними елементами центральної частини є центральний процесор і оперативна пам'ять.

Окрім ОП у всіх ПК є внутрішня постійна пам'ять для зберігання постійних даних і програм. ОП – до недавнього часу доволі цінна частина апаратури. Об'єм її обмежений, тому великі масиви або таблиці інформації зберігаються у зовнішніх запам'ятовуючих пристроях. До них відносять: накопичувачі на магнітних дисках НГМД, НЖМД, пам'ять типу FLESH, накопичувачі на магнітних стрічках, накопичувачі на оптичних і магнітооптичних дисках.

У сучасних ПК реалізована *віртуальна пам'ять*, яка дозволяє користувачеві працювати з розширеним простором пам'яті комп'ютера. Віртуальна пам'ять будується на основі оперативної пам'яті і зовнішнього запам'ятовуючого пристрою, а також комплексу програмно-апаратних засобів, що забезпечують динамічну переадресацію даних, внаслідок чого користувач не піклується про своєчасне передання інформації із ЗЗП в ОП. Функції з необхідного переміщення інформації бере на себе обчислювальна система.

Зовнішні запам'ятовуючі пристрої ЗЗП, пристрої введення і пристрої виведення утворюють периферійну частину ПК. Склад пристроїв може сильно відрізнятися в різних екземплярах, тому говорять про конфігурацію ПК.

Продуктивність і ефективність використання ПК визначається не тільки процесором і ОП, але великою мірою й технічними даними периферійних пристроїв, а також особливостями їх спільної роботи з центральною частиною ПК. Зв'язок між центральною і периферійною частинами здійснюється за допомогою сполучень, які називаються **інтерфейсами**.

Інтерфейс – це сукупність стандартизованих апаратних і програмних засобів, що забезпечують обмін інформацією між

пристроями. В основі побудови інтерфейсів лежить уніфікація і стандартизація (використання єдиних способів кодування даних, форматів даних, використання єдиних роз'ємів і т. ін.). Наявність інтерфейсів дозволяє уніфікувати передання даних між пристроями, незалежно від їх особливостей. Особливості враховують контролери – спеціальні блоки ПК для управління роботою периферійних пристроїв.

Усі моделі сучасних ПК мають *магістрально-модульний тип архітектури*. При такій структурі всі пристрої ПК обмінюються інформацією і сигналами через системну шину або магістраль. Фізично вона є системою функціонально об'єднаних дротів, якими передається 3 потоки даних:

- безпосередньо інформація (дані);
- управляючі сигнали;
- адреси.

Модульність дозволяє міняти пристрої (модулі) ПК, до магістралі можуть підключатися додаткові периферійні пристрої, одні моделі пристроїв можуть замінюватися іншими. Можливе збільшення внутрішньої пам'яті, заміна мікропроцесора на більш продуктивний.

Апаратне підключення периферійного пристрою до магістралі здійснюється через контролер. Контролер або адаптер – це електронна схема, яка забезпечує управління роботою пристроїв комп'ютера. Програмне управління роботою пристрою здійснюється за допомогою програми-драйвера, яка є складовою частиною операційної системи.

Модульна організація спирається на магістральний принцип обміну інформацією між пристроями ПК із системною шиною (див. рис. 2.3).

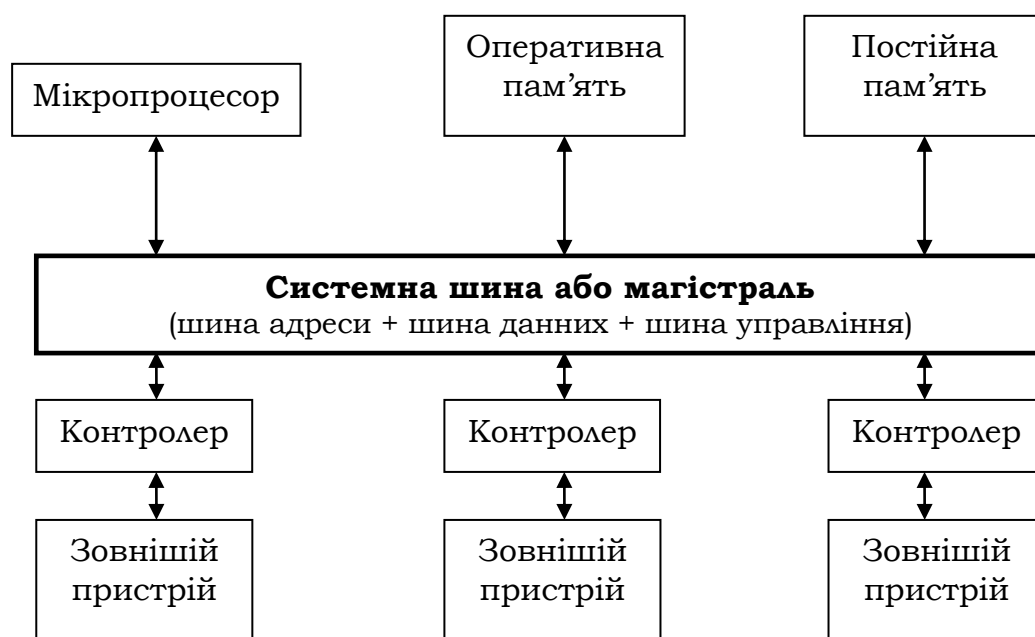


Рис. 2.3. Схема обміну інформацією за допомогою системної шини

Кількість дротів у системній шині, призначених для передавання інформації, називається **розрядністю шини**.

Оскільки шина є загальною, використовуватися вона може в кожен певний момент часу тільки одним яким-небудь пристроєм. Для визначення порядку використання шини передбачена система переривань, яка віддає шину для використання пристрою з найбільшим пріоритетом.

2.5. АРХІТЕКТУРНА ОРГАНІЗАЦІЯ ПРОЦЕСОРА ЕОМ

Під **архітектурною організацією**, або **архітектурою процесора**, будемо розуміти функціональні можливості його апаратних складових мікропроцесора, які використовуються для представлення даних, машинних операцій і процесів обчислень.

Розробляючи програмне забезпечення для цифрового пристрою, програміст повинен знати його архітектурні особливості і технічні характеристики. Ця вимога необхідна при використанні мови асемблера, також вона може виявитися істотною і при використанні мови високого рівня. Однак рівень знань програміста щодо архітектури, і тим більше щодо схемотехнічних елементів МПС не є настільки детальним, як у інженера – розробника мікропроцесорних пристроїв. На відміну від останнього, програміст може мати потребу в з'ясуванні і розумінні лише елементів і характеристик МПС, які явно відбиваються в програмах і (або) повинні бути враховані при розробленні і виконанні програм. До таких елементів і характеристик варто віднести: кількість та імена програмно доступних регістрів; розрядність машинного слова; систему команд; доступний розмір, типи і адреси запам'ятовуючих пристроїв; схему оброблення переривань; способи адресації пам'яті і зовнішніх пристроїв. Сукупність подібних відомостей називають **програмною моделлю** процесора та використовують при створенні програмного забезпечення цифрового пристрою.

Процесор займає в архітектурі ЕОМ центральне місце, здійснюючи управління взаємодією всіх основних компонентів, що входять до складу ЕОМ. Він безпосередньо здійснює оброблення інформації і програмне управління цим процесом, дешифрує і виконує команди програм, організує звернення до оперативної пам'яті (ОП), у потрібних випадках ініціює операції введення/виведення і роботу зовнішніх пристроїв ЗП, сприймає і обробляє запити, що поступають як від внутрішніх пристроїв ЕОМ, так і із зовнішнього середовища (організація системи переривань).

У сучасних процесорах часто використовується **принцип мікропрограмного управління**. Виконання кожної команди

складається з виконання дрібніших операцій – мікрокоманд, що виконують певні елементарні дії. Набір мікрокоманд визначається системою команд і логічною структурою конкретної ЕОМ. У деяких ЕОМ (у першу чергу, спеціалізованих) усі або частина команд реалізуються апаратно, що дозволяє підвищувати їх продуктивність за рахунок втрати певної частини гнучкості системи команд машини. Як апаратний, так і програмний спосіб реалізації команд ЕОМ має свої плюси і мінуси.

Усю ієрархію мов для опису обчислювальних процесів в ЕОМ можна поділити на чотири рівні: 1) булеві операції (функціонування комбінаційних схем); 2) мікрокоманди (функціонування вузлів ЕОМ); 3) команда (функціонування ЕОМ); 4) оператор мови високого рівня (опис алгоритму задачі, яка вирішується). Мова програмування мікропроцесора призначена для опису цифрових пристроїв на рівні регістрів. Вона має прості та наочні засоби опису машинних слів, регістрів, шин та інших базових елементів ЕОМ. Для встановлення часового співвідношення між виконанням мікрокоманд визначається одиниця часу (такт), впродовж якої виконується найдовша мікрокоманда. Такти генеруються спеціальним пристроєм – генератором тактової частоти. Тактова частота вимірюється в МГц або ГГц та значною мірою визначає швидкість ЕОМ. Разом із тактовою частотою швидкість визначають такі фактори, як:

- час доступу до пам'яті;
- час вибірки;
- розрядність;
- архітектура процесора і його співпроцесорів.

Регістр процесора – це комірка швидкодіючої внутрішньої пам'яті процесора, яка використовується для тимчасового збереження операндів, з якими безпосередньо проводяться обчислення, та даних з метою швидкого доступу до них. Крім того, у регістрах зберігається і додаткова інформація, яка потрібна процесору для функціонування (зокрема, поточний стан процесора, адреса наступної команди тощо).

Процесори зазвичай містять від кількох до кількох сот регістрів різного функціонального призначення.

Регістри характеризуються своєю розрядністю, тобто кількістю біт інформації, яка може в них розміщуватись. Наприклад: 8-розрядний регістр, 64-розрядний регістр. Регістри найчастіше реалізуються як масиви статичної пам'яті з довільним доступом (SRAM), хоча також можуть реалізовуватись на базі бістабільних комірок (тригерів), інших схемотехнічних рішень, придатних для реалізації у швидких запам'ятовуючих пристроях.

З точки зору архітектури ЕОМ, під терміном «реєстри процесора» розуміють лише набір тих реєстрів, які доступні програмісту в рамках документованої програмної моделі процесора. Більш точно такі реєстри називають архітектурними. Наприклад, архітектура процесорів типу Intel 80x86 визначає лише вісім 32-розрядних реєстрів загального призначення, але процесор фактично містить набагато більше реальних апаратних реєстрів. Така надлишковість потрібна для реалізації деяких мікроархітектурних оптимізацій швидкодії процесора.

За функціональним призначенням реєстри процесора поділяються на такі:

1) *реєстри загального призначення* – можуть зберігати і дані, і адреси;

2) *реєстри даних* – використовуються для збереження даних. До них відносяться реєстри цілочисельних даних, реєстри констант, реєстри рухомої коми, векторні реєстри. Реєстри рухомої коми зберігають дані для обчислень з рухомою комою. Реєстри констант зберігають константи. Векторні реєстри зберігають векторні дані та забезпечують векторні обчислення (наприклад, у мультимедійних розширеннях архітектури Intel 80x86);

3) *адресні реєстри* – зберігають адреси (номери комірок) у пам'яті та використовуються в операціях з пам'яттю. Такі реєстри іноді називаються індексними, або базовими;

4) *реєстри спеціального призначення* – зберігають внутрішню інформацію, необхідну для функціонування процесора (лічильник команд, реєстр стану процесора та ін.).

Розглянемо укрупнені схеми центрального процесора (ЦП) деякої формальної ЕОМ (див. рис. 2.4), де зображені тільки основні його блоки: управляючі реєстри (УР), пристрій управління (ПУ), постійний запам'ятовуючий пристрій (ПЗП), арифметико-логічний пристрій (АЛП), реєстри загального призначення, або реєстрова пам'ять (РЗП), кеш-пам'ять та інтерфейсний блок (ІБ). Разом із перерахованими вище ЦП містить ряд інших блоків (переривання, захисту ОП, контролю і діагностики та ін.), структура і призначення яких тут не розглядаються.

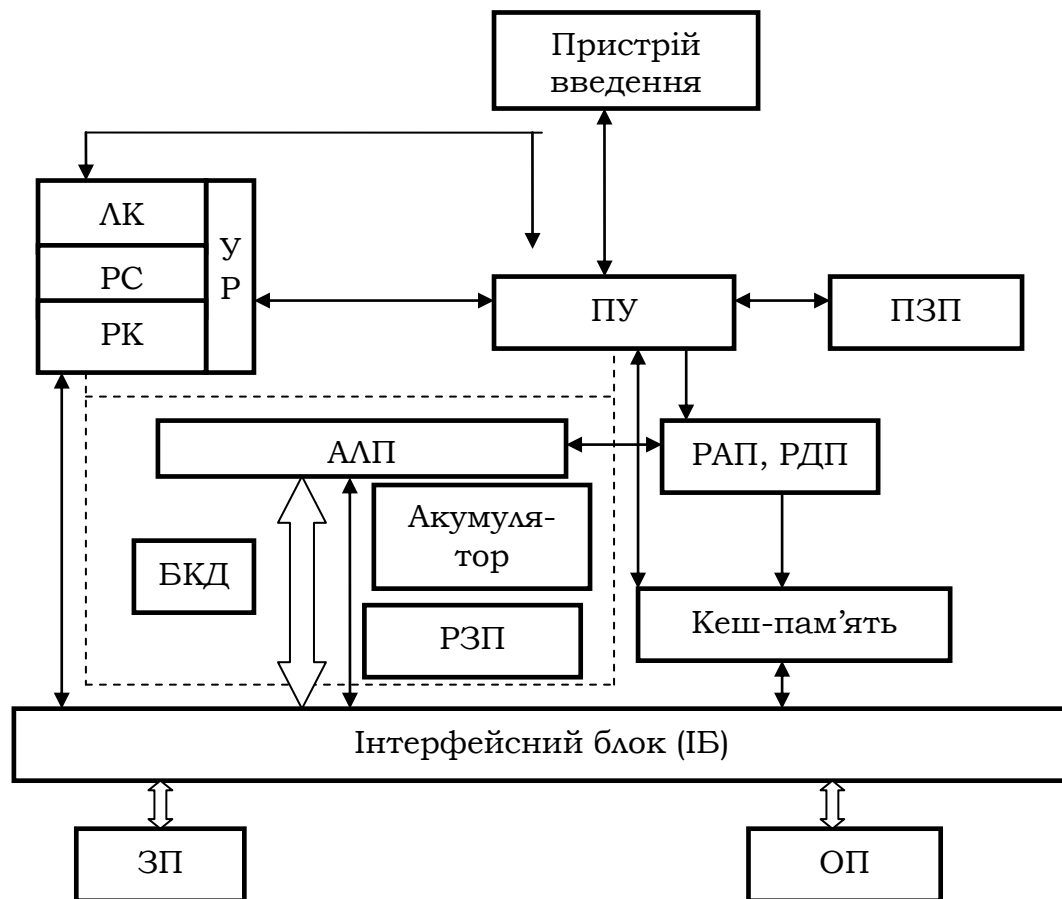


Рис. 2.4. Укрупнена схема центрального процесора ЕОМ:

ЛК – лічильник команд; РС – регістр стану; РК – регістр команд;
 УР – управляючі регістри; ПУ – пристрій управління; ПЗП – постійний запам'ятовуючий пристрій; АЛП – арифметико-логічний пристрій;
 РАП – регістр адреси пам'яті; РДП – регістр даних пам'яті; БКД – блок контролю та діагностики; РЗП – регістри загального призначення;
 ЗП – зовнішній пристрій; ОП – оперативна пам'ять

Блок ПУ виробляє послідовність управляючих сигналів, ініціює виконання відповідної послідовності мікрокоманд (що знаходяться в ПЗП), що реалізовує поточну команду. Разом із цим ПУ координує функціонування всіх пристроїв ЕОМ за допомогою відсилання управляючих сигналів: обмін даними між ЦП і ОП, зберігання і оброблення інформації, інтерфейс з користувачем, тестування і діагностика та ін. Тому ПУ доцільно розглядати як окремий блок ЦП. Проте на практиці більшість управляючих схем, розподілені по всій ЕОМ. Вони зв'язані великим числом управляючих ліній, по цих лініях передаються сигнали для синхронізації операцій у всіх пристроях ЕОМ і тих, що приймають сигнали про їх стан.

Блок управляючих регістрів (УР) призначений для тимчасового зберігання інформації, яка необхідна для керування роботою ЕОМ, і містить регістри і лічильники, що беруть участь спільно з ПУ в управлінні обчислювальним процесом: регістр стану (РС), регістр команд (РК), лічильник команд (ЛК).

Лічильником команд (ЛК) є реєстр, що зберігає адресу в ОП виконуваної команди: у період виконання поточної команди його вміст оновлюється на адресу наступної команди.

Реєстр команд (РК) містить виконувану команду (його виходи пов'язані з управляючими схемами, генерують розподілені в часі сигнали, необхідні для виконання команд).

Реєстр стану (РС) має побітну організацію – кожний біт зберігає відповідну ознаку отриманого результату операції АЛУ.

Блок реєстрів загального призначення (РЗП) містить реєстри надоперативної пам'яті (зі швидкодією, більшою ніж ОП) невеликого об'єму, що дозволяють підвищити швидкодію і продуктивність ЦП. Ці реєстри використовуються в командах мікропроцесора для скороченої реєстрової адресації (вказуються тільки номери реєстрів) і слугують для зберігання операндів, результатів операцій як базові і індексні реєстри, покажчики стека та ін. У деяких ЦП базові та індексні реєстри входять до складу блока УР.

Блок АЛП слугує для виконання арифметичних і логічних операцій над даними, що поступають з ОП і що зберігаються в РЗП, і працює під управлінням ПУ. АЛП виконує арифметичні операції над бінарними числами з фіксованою і рухомою комою, над десятковими числами, проводить оброблення символічної інформації над словами фіксованої і змінної довжини. Логічні операції проводяться над окремими бітами, групами бітів, байтами і їх послідовностями. Тип операції, яку виконує АЛП, визначається поточною командою програми, що функціонує в даний момент, точніше, АЛП слугує для виконання будь-якої операції, яку завдає йому ПУ. У загальному випадку інформація, яку обробляє ЕОМ, складається зі слів, що містять фіксоване число n бітів (наприклад, $n = 8, 16, 32, 64, 128$ біт). У цьому випадку АЛП повинен мати можливість проводити операції над n -бітними словами, операнди поступають із ОП на реєстри ЦП, а ПУ указує операцію, яку необхідно над ними провести, результат кожної арифметико-логічної операції зберігається в спеціальному реєстрі – акумуляторі.

Акумулятор є головним реєстром ЦП і входить до блока РЗП. В акумуляторі завжди зберігається результат останньої операції, яку виконав АЛП. Через акумулятор здійснюється обмін даними між запам'ятовуючим пристроєм та ЦП.

Конструктивно АЛП виконується на одній або кількох ВІС/НВІС, при цьому ЦП може мати один АЛП універсального призначення або кілька спеціалізованих для окремих видів операцій. В останньому випадку збільшується структурна складність ЦП, але підвищується його швидкодія за рахунок

спеціалізації і спрощення схем обчислення окремих операцій. Такий підхід широко використовується в сучасних ЕОМ загального призначення і суперЕОМ для підвищення їх продуктивності. Попри різні класи ЕОМ, їх АЛП використовують загальні принципи виконання арифметико-логічних операцій. Відмінності стосуються вирішень схемотехнічних задач щодо організації АЛП і принципів реалізації операцій, що забезпечують прискорення їх виконання.

Інтерфейсний блок (ІБ) забезпечує обмін інформацією між ЦП та ОП і захист комірок ОП від несанкціонованого для поточної програми доступу, а також зв'язок ЦП із зовнішніми пристроями та іншими зовнішніми по відношенню до ЦП пристроями, у якості яких можуть виступати інші процесори і ЕОМ. Зокрема, ІБ містить два реєстри, що забезпечують зв'язок з ОП: реєстр адреси пам'яті (РАП) і реєстр даних пам'яті (РДП). Перший реєстр використовується для зберігання адреси комірки ОП, з якою проводиться обмін даними, а другий містить власне дані для обміну. Блок контролю і діагностики (БКД) призначений для виявлення збоїв і відмов вузлів ЦП, відновлення роботи поточної програми після збоїв і локалізації несправностей при відмовах.

З урахуванням сказаного представимо загальну *схему виконання програми процесором*. Виконання програми, що знаходиться в ОП, починається з того, що в ЛК засилається адреса першої її команди, вміст ЛК пересилається в РАП і в ОП посилається сигнал управління прочитуванням. Слово (у даному випадку перша команда програми), що через деякий час (відповідно часу доступу до ОП) адресується, витягується з ОП і завантажуються в РДП, потім вміст РДП пересилається в РК. На цій стадії команда готова для декодування її в ПУ і виконання. Якщо команда містить операцію, яка повинна бути виконана АЛП, то слід отримати необхідні операнди. Якщо операнд знаходиться в ОП (а він може бути також в УР), його необхідно вибрати з пам'яті. Для цього в РАП пересилається адреса операнда і починається цикл читання. Операнд, вибраний із пам'яті в РДП, може бути переданий в АЛП. Вибравши таким чином один або декілька операндів, АЛП може виконати необхідну операцію, зберігши її результат в одному з РЗП. Якщо результат операції необхідно запам'ятати в ОП, він повинен бути надісланий у РДП. Адреса комірки пам'яті, у яку необхідно помістити результат, пересилається в РАП, і починається цикл запису. Тим часом вміст ЛК збільшується, указуючи адресу наступної команди, яка повинна виконуватися. Таким чином, як тільки завершиться виконання поточної команди, може відразу ж початися вибірка наступної команди програми.

Крім передання даних між ОП і ЦП, необхідно забезпечити обмін даними із зовнішніми пристроями, при виконанні машинних команд, які забезпечують процедури введення/виведення.

Природний порядок виконання програм може порушуватися під час отримання сигналу переривання. Переривання є вимогою на обслуговування, яке здійснюється ЦП, що виконує відповідну програму оброблення переривання. Оскільки переривання і його оброблення можуть змінити внутрішній стан ЦП, то воно зберігається в ОП перед початком роботи програми. Збереження стану досягається пересилкою вмісту УР і деякої інформації, необхідної для управління роботою ЦП, в ОП. Після завершення програми оброблення переривання стан ЦП відновлюється, дозволяючи продовжити виконання перерваної програми.

2.6. ПРИНЦИПИ ОРГАНІЗАЦІЇ СИСТЕМИ ПЕРЕРИВАНЬ ПРОГРАМ

Під час виконання ЕОМ поточної програми усередині машини або у зовнішньому середовищі (наприклад, у технологічному процесі, яким керує ЕОМ) можуть виникати події, що вимагають негайної реакції на них з боку машини. Реакція полягає в тому, що машина перериває оброблення поточної програми і переходить до виконання деякої іншої програми, спеціально призначеної для даної події. По завершенні цієї програми ЕОМ повертається до виконання перерваної програми. Розглянутий процес називається **перериванням програм**. Принципово важливим є те, що моменти виникнення подій, що вимагають переривання програм, заздалегідь невідомі і тому не можуть бути враховані при програмуванні. Кожна подія, що вимагає переривання, супроводжується сигналом, який сповіщає процесор про подію. Назвемо ці сигнали **запитами на переривання**. Програму, до виконання якої переходить машина, назвемо **перериваючою програмою**.

Запити на переривання можуть виникати усередині самої ЕОМ і в її зовнішньому середовищі. До першого відносяться, наприклад, запити при виникненні в ЕОМ таких подій, як помилка в роботі її апаратури, переповнення розрядної сітки, спроба ділення на 0, вихід із установленної для даної програми області пам'яті, запит від периферійного пристрою на операцію введення/виведення, завершення операції введення/виведення периферійним пристроєм або виникнення під час цієї операції особливої ситуації та ін. Хоча деякі із зазначених подій породжуються самою програмою, моменти їх появи, як правило, неможливо передбачити. Запити із зовнішнього середовища можуть виникати від інших ЕОМ, датчиків та пристроїв.

Для обслуговування переривань в ЕОМ необхідно додати відповідні апаратурні і програмні засоби, сукупність яких одержала назву системи переривань програм або контролера переривань. Основними функціями системи переривань є:

- запам'ятовування стану програми, що переривається, і здійснення переходу до перериваючої програми;
- відновлення стану перерваної програми і повернення до неї.

При наявності декількох джерел запитів переривання має бути встановлений певний порядок (**дисципліна**) в обслуговуванні запитів, що надходять. Іншими словами, між запитами (і відповідними перериваючими програмами) повинні бути встановлені пріоритетні співвідношення, що визначають, який із декількох запитів, що надійшли, підлягає обробленню в першу чергу і чи має право даний запит (перериваюча програма) перервати ту або іншу програму.

Глибина переривання – це максимальне число програм, що можуть переривати одна одну. Якщо після переходу до перериваючої програми і аж до її закінчення прийом інших запитів забороняється, то говорять, що система має глибину переривання, рівну 1. Глибина дорівнює n , якщо допускається послідовне переривання до n програм. Глибина переривання звичайно збігається з числом рівнів пріоритету в системі переривань. Системи з великим значенням глибини переривання забезпечують більш швидку реакцію на термінові запити. Глибина переривання відрізняється в ЕОМ різних типів. Наприклад, вектор стану мікропроцесора K580ИК80 містить у собі зміст 16-розрядного лічильника команд (адреса чергової команди), зміст 8-розрядного регістра ознак, названий у документації на цей мікропроцесор **словом стану процесора**, і зміст 8-розрядного акумулятора АЛП. Більш великі ЕОМ мають більш складні структури вектора стану, або, інакше кажучи, слова стану програми.

Використання слова (вектора) стану – поширений прийом побудови керування пристроями обчислювальної техніки. У багатьох пристроях ЕОМ для організації їхнього функціонування формуються свої, специфічні слова стану (або байти стану), що фіксують у вигляді деякого коду стану пристрою, наприклад, готовність його до виконання операції, що задається, успішне або неуспішне завершення операції і т. п.

Вектор переривання містить усю необхідну інформацію для переходу до перериваючої програми, у тому числі її початкову адресу. Кожному запиту (рівню) переривання, а в ряді випадків і кожному периферійному пристрою, відповідає свій вектор переривання, здатний ініціювати виконання відповідної

перериваючої програми. Вектор переривання звичайно знаходиться в спеціально виділених фіксованих комірках пам'яті.

Поріг переривання – це метод, який дозволяє в ході обчислювального процесу програмним шляхом змінювати рівень пріоритету переривання, іншими словами, задавати поріг переривання, тобто мінімальний рівень пріоритету запитів, яким дозволяється переривати програму, що виконується процесором.

Поріг переривання задається командою програми, що встановлює в регістрі порога переривань код порога переривання. Спеціальна схема виділяє найбільш пріоритетний запит переривання, порівнює його пріоритет із порогом переривання і, якщо він виявляється вище порога, виробляє загальний сигнал переривання – починається процедура переривання.

Маска переривання являє собою двійковий код, розряди якого представлені у відповідності запитам або класам переривання. Маска завантажується командою програми в регістр маски. Стан 0 у розряді регістра маски дозволяє, а стан 1 – забороняє (маскує) переривання поточної програми від перепідключення ліній сигналів запитів на входах системи переривань або у більшості систем переривань шляхом програмної заміни розподілу рівнів між входами.

2.7. ПОНЯТТЯ ПРО СТАН ПРОЦЕСОРА (ПРОГРАМИ).

ВЕКТОР (СЛОВО) СТАНУ

При виконанні процесором програми після кожного робочого такту, а тим більше в результаті завершення виконання чергової команди, змінюється вміст регістрів, лічильників, стан окремих прапорів. Можна говорити, що змінюється стан процесора або, вживаючи іншу термінологію, **стан програми**.

Поняття **стану процесора (стану програми)** займає важливе місце в організації обчислювального процесу в ЕОМ.

Інформація про стан процесора (програми) є основою багатьох процедур управління обчислювальним процесом, наприклад при аналізі ситуацій при відмовах і збоях, при відновленні виконання програми після переривання, для фіксації стану процесора (програми) у момент переходу в мультипрограмному режимі від оброблення даної програми до іншої і т. ін.

Станом процесора (програми) після даного такту або після виконання даної команди, строго кажучи, слід уважати сукупність станів у відповідний момент часу всіх елементів пристрою, які запам'ятовують – тригерів, регістрів, елементів пам'яті.

Проте не вся ця інформація зникає або спотворюється при переході до чергової команди або іншої програми. Тому зі всього

різноманіття інформації про стан процесора (програми) відбираються найбільш істотні її елементи, як правило, ті, що змінюються при переході до іншої команди або програми.

Сукупність значень цих інформаційних елементів отримала назву **вектора стану** або слова стану процесора (програми).

Вектор стану в кожний момент часу зберігає інформацію, достатню для продовження програми або перезапуску програми з моменту формування даного вектора стану. Інша інформація, яка також характеризує стан процесора (наприклад, вміст регістрів), може бути відновлена програмним шляхом за копією, яка зберігається в пам'яті.

Вектор стану формується у відповідному регістрі (регістрах) процесора, зазнаючи зміни після виконання кожної команди.

Сукупності інформаційних елементів, із яких складаються вектори стану, відрізняються в ЕОМ різних типів. Найпростіше слово стану виглядає у мікропроцесорів. Наприклад, вектор стану мікропроцесора K580 (див. рис. 2.5) включає вміст 16-розрядного лічильника команд (адреса чергової команди), вміст 8-розрядного регістра ознак і вміст 8-розрядного акумулятора АЛП. Сучасні ЕОМ мають складніші структури вектора стану.

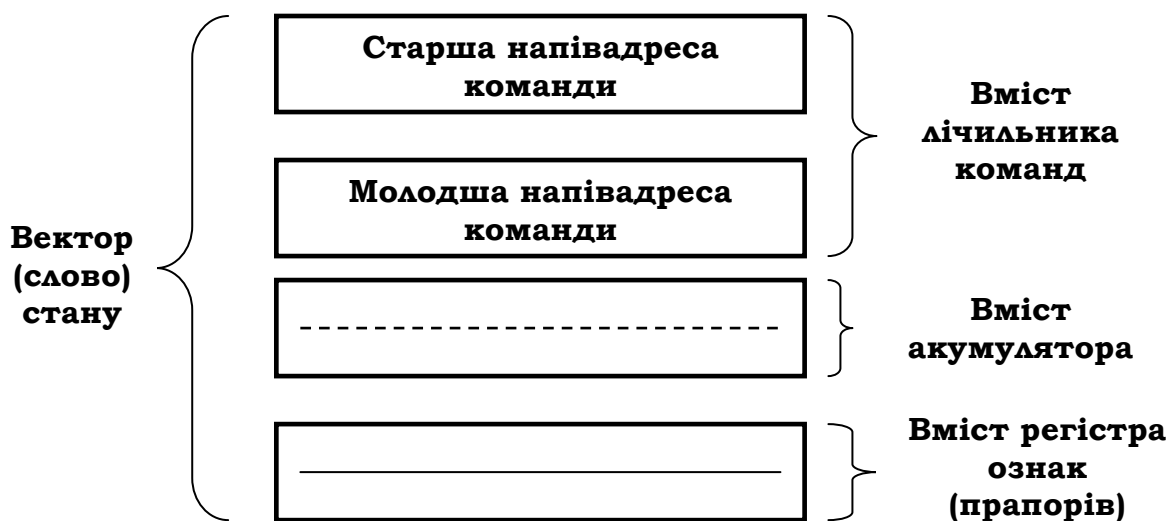


Рис. 2.5. Вектор стану 8-розрядного мікропроцесора K580 (чотири 8-розрядних слова)

Використання слова (вектора) стану – поширений прийом побудови управління пристроями обчислювальної техніки. У багатьох пристроях ЕОМ для організації їх функціонування формуються свої специфічні слова стану (або байти стану), що фіксують у вигляді деякого коду стан пристрою, наприклад готовність його до виконання операції, що задається, успішне або неуспішне завершення операції і т. ін.

2.7. ЗАГАЛЬНІ ВІДОМОСТІ І КЛАСИФІКАЦІЯ ПРИСТРОЇВ ПАМ'ЯТІ

Пам'яттю ЕОМ називається сукупність пристроїв для запам'ятовування, зберігання і видачі інформації. Окремі пристрої, що входять в цю сукупність, називають **запам'ятовуючими пристроями** (ЗП), або пам'яттю, того або іншого типу.

Продуктивність і обчислювальні можливості ЕОМ значною мірою визначаються складом і характеристиками її ЗП. У складі ЕОМ використовується одночасно кілька типів ЗП, які відрізняються принципом дії, характеристиками і призначенням.

Основними операціями з пам'яттю є занесення інформації в пам'ять – **запис** і вибірка інформації з пам'яті – **читання**. Обидві ці операції називаються зверненням до пам'яті.

При зверненні до пам'яті проводиться читання або запис деякої одиниці даних – різної для пристроїв різного типу. Такою одиницею може бути, наприклад, байт, машинне слово або блок даних.

Найважливішими характеристиками окремих пристроїв пам'яті є ємкість, питома ємкість та швидкодія.

Ємкість пам'яті визначається максимальною кількістю даних, які можуть у ній зберігатися.

Питома ємкість є відношення ємкості ЗП до його фізичного об'єму.

Щільність запису є відношення ємкості ЗП до площі носія. Наприклад, у HDD ємкістю до 10 Гб щільність запису складає 2 Гбіт на кв. дюйм.

Швидкодія пам'яті визначається тривалістю операції звернення, тобто часом, що витрачається на пошук потрібної одиниці інформації в пам'яті та на її читання (час звернення при читанні), або часом на пошук місця для запису одиниці інформації і на її запис у пам'ять (час звернення при записі).

У деяких пристроях пам'яті читання інформації супроводжується її руйнуванням (стиранням). У такому разі цикл звернення повинен містити операцію відновлення (регенерації) інформації на колишньому місці в пам'яті.

Тривалість звернення до пам'яті (час циклу пам'яті) при читанні:

$$T_{\text{чзверн}} = T_{\text{чдост}} + T_{\text{чит}} + T_{\text{реген}}, \quad (4)$$

де $T_{\text{чзверн}}$ – тривалість звернення до пам'яті; $T_{\text{чдост}}$ – час доступу при читанні, який визначається з моменту початку операції звернення до моменту, коли стає можливим доступ до даної одиниці інформації; $T_{\text{чит}}$ – тривалість самого фізичного процесу читання, тобто процесу виявлення і фіксації станів відповідних запам'ятовуючих елементів або комірок поверхні носія інформації; $T_{\text{реген}}$ – час, необхідний на регенерацію одиниці інформації.

Тривалість звернення (час циклу) при записі:

$$T_{\text{зверн}} = T_{\text{дост}} + T_{\text{підг}} + T_{\text{зап}}, \quad (5)$$

де $T_{\text{зверн}}$ – тривалість звернення при записі; $T_{\text{дост}}$ – час доступу при записі, тобто час від моменту звернення до моменту, коли стає можливим доступ до запам'ятовуючих елементів (або комірок поверхні носія), у які проводиться запис; $T_{\text{підг}}$ – час підготовки, що витрачається на приведення в початковий стан запам'ятовуючих елементів або комірок поверхні носія інформації для запису певної одиниці інформації (наприклад, байта або слова); $T_{\text{зап}}$ – час запису інформації, тобто зміни стану запам'ятовуючих елементів.

Переважно $T_{\text{чзверн}} = T_{\text{дост}}$.

Як тривалість циклу звернення до пам'яті ($T_{\text{дост}}$) приймається величина:

$$T_{\text{дост}} = \max(T_{\text{чзверн}}, T_{\text{дост}}).$$

Залежно від операцій звернення розрізняють: а) пам'ять із довільним доступом (можливе як читання, так і запис даних у пам'ять); б) пам'ять тільки для читання інформації («постійна», або «одностороння»). Запис інформації в постійну пам'ять проводиться в процесі її виготовлення або прошивки.

Ці типи пам'яті відповідають термінам RAM (*random access memory* – пам'ять з довільним доступом) і ROM (*read only memory* – пам'ять тільки для читання).

За способом організації доступу розрізняють пристрої пам'яті з безпосереднім (довільним), з прямим (циклічним) і послідовним доступами.

У пам'яті з *безпосереднім (довільним)* доступом час доступу і цикл звернення не залежать від місця розташування комірки пам'яті, з якої проводиться читання або в яку записується інформація. У більшості випадків безпосередній доступ реалізується за допомогою електронних (напівпровідникових) ЗП. У таких пристроях цикл звернення зазвичай складає 70 і менше наносекунд. Кількість розрядів, що читаються або записуються в пам'ять паралельно за одну операцію звернення, називаються **шириною вибірки**.

В інших типах пам'яті використовуються повільніші електромеханічні процеси. У пристроях *пам'яті з прямим доступом*, до яких відносяться дискові накопичувачі, завдяки безперервному обертанню носія інформації можливість звернення до певної комірки пам'яті для читання або запису циклічно повторюється. У такій пам'яті час доступу складає зазвичай від кількох часток секунди до кількох десятків мілісекунд.

У пам'яті з *послідовним доступом* здійснюється послідовний перегляд комірок носія інформації. Характерним прикладом ЗП

такого типу є магнітні стрічки та стримери (*streamer*). Час доступу в таких ЗП може доходити до кількох хвилин. Пристрої з такою організацією вважаються застарілими.

Вимоги до ємкості і швидкодії пам'яті є суперечливими. Чим більше швидкодія, тим технічно складніше і дорожче обходиться збільшення ємкості пам'яті. Ціна ЗП складає значну частину загальної ціни ЕОМ. Тому пам'ять ЕОМ організовується у вигляді ієрархічного набору ЗП, які мають різну швидкодію і ємкість. Ієрархічна структура пам'яті дозволяє економно поєднувати зберігання великих об'ємів інформації зі швидким доступом до інформації в процесі оброблення.

ЗП розрізняються також за функціональним призначенням в архітектурі ЕОМ, тобто в структурі ЕОМ розрізняють декілька типових ЗП.

Оперативною, або основною, пам'яттю (ОП) називають пристрій, який призначений для зберігання інформації (даних, програм, проміжних і кінцевих результатів оброблення), що безпосередньо використовується в процесі виконання операцій арифметико-логічним пристроєм (АЛП) і пристроєм управління (ПУ) процесора. У процесі оброблення інформації здійснюється тісна взаємодія процесора і ОП. Із ОП у процесор поступають команди програми і операнди, над якими проводяться передбачені командою операції, а з процесора в ОП прямують проміжні і кінцеві результати оброблення. Характеристики ОП безпосередньо впливають на основні показники ЕОМ і, у першу чергу, на швидкість її роботи. Пристрої ОП виготовляються на інтегральних мікросхемах з великим ступенем інтеграції (напівпровідникові).

У ряді випадків швидкодія ОП виявляється недостатньою, і до складу ЕОМ включають надоперативну пам'ять НОП (буферну або кеш-пам'ять на кілька сотень або тисяч кілобайт з циклом звернення, який складає кілька наносекунд). Швидкодія НОП повинна відповідати швидкості роботи арифметико-логічного пристрою процесора. Надоперативна (буферна) пам'ять використовується для проміжного зберігання ділянок програм і даних, прочитаних процесором з ОП, а також для зберігання робочих параметрів програми, значень індексних регістрів, службової інформації щодо управління обчислювальним процесом. Вона виконує роль проміжної ланки між швидкодіючими логічними пристроями процесора і повільнішою ОП. Пристрої ОП і НОП є швидкодіючі ЗП із довільним і безпосереднім доступом.

Зазвичай ємкість ОП виявляється недостатньою для зберігання всіх необхідних даних в ЕОМ. Тому ЕОМ містить у своєму

складі декілька ЗП із прямим доступом на жорстких дисках (ЗП на HDD).

Оперативна пам'ять разом із НОП і деякими іншими спеціалізованими видами пам'яті (мається на увазі постійна пам'ять, яка буде розглянута пізніше) процесора утворюють *внутрішню пам'ять* ЕОМ. Електромеханічні ЗП утворюють зовнішню пам'ять ЕОМ, а тому вони називаються зовнішніми запам'ятовуючими пристроями ЗЗП.

ЗП будь-якого типу складається з масиву комірок, що запам'ятовують інформацію, і блоків управління, які слугують для пошуку в масиві необхідних даних, а також для запису і читання (а в ряді випадків і для регенерації) інформації.

Розглянемо організаційні принципи роботи ЗП, які розрізняють як адресну, асоціативну і стекову організацію пам'яті.

Адресна, асоціативна і стекова організації пам'яті

ЗП з довільним доступом, як правило, містить безліч однакових запам'ятовуючих елементів, які утворюють запам'ятовувачий масив (ЗМ). ЗМ розділений на окремі комірки: кожна з них призначена для зберігання двійкового коду, число розрядів у якому визначається шириною вибірки пам'яті (зокрема, це може бути одне, половина або кілька машинних слів). Спосіб організації пам'яті залежить від методів розміщення і пошуку інформації в ЗМ. За цією ознакою розрізняють адресну, асоціативну і стекову (магазинну) пам'ять.

Адресна пам'ять. У пам'яті з адресною організацією розміщення і пошук інформації в ЗМ засновані на використанні адреси зберігання слова даних (числа, команди і т. ін.). Адресою слугує номер комірки ЗМ, у якій це слово розміщується. Запис (або читання) слова даних у ЗМ здійснюється за командою, яка ініціює цю операцію. У команді вказується адреса (номер комірки), у яку проводиться запис (або читання).

Типова структура адресної пам'яті містить масив з n -розрядних комірок і його апаратне оточення, що включає регістр адреси (РгА) пам'яті, що має k розрядів, регістр даних (РгД), блок адресної вибірки (БАВ), блок підсилювачів прочитання (БПП), блок порозрядних підсилювачів-формуваців сигналів запису (БПЗ) і блок управління пам'яттю (БУП).

Схему адресної пам'яті показано на рис. 2.6. Код адреси зберігається в РгА. Блок адресної вибірки (БАВ) формує у відповідному елементі пам'яті сигнали, що дозволяють здійснити в комірці читання або запис слова.

Цикл звернення до пам'яті ініціюється надходженням у блок управління пам'яттю (БУП) сигналу «Звернення». Загальна частина циклу «Звернення» складається з прийому в регістр адреси

РгА з шини адреси (ША) *адреси звернення* та прийому і розшифровки сигналу «Операція», який вказує тип операції (читання або запис). Далі при прочитуванні БАВ дешифрує адресу, поси- лає сигнали прочитування в задану адресою комірку ЗМ, при цьому код записаного в комірці слова прочитується підсилюва- чами прочитування БПП і передається в РгД. Операція прочи- тування завершується видачею слова з РгД на вихідну шину даних ШДвих.

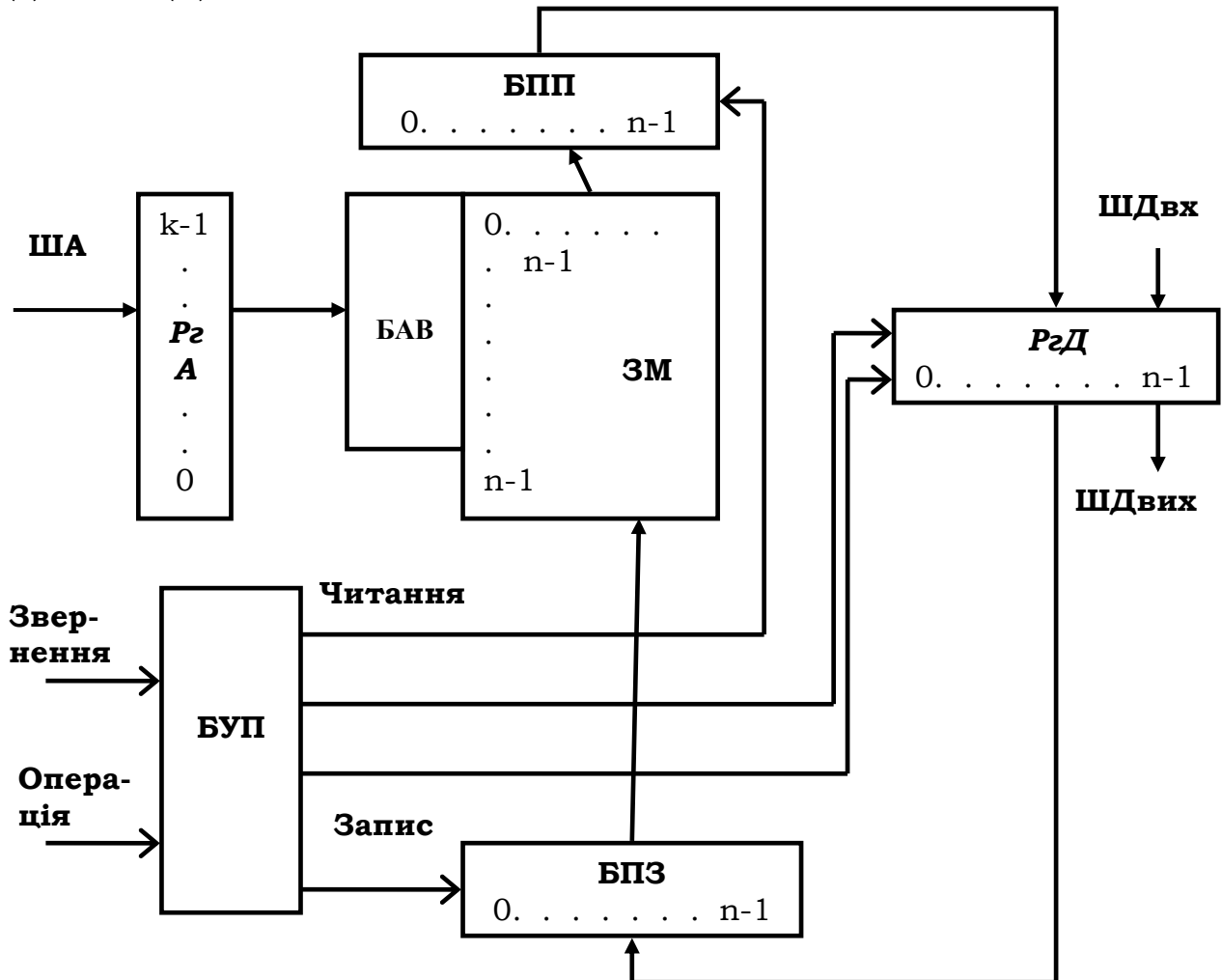


Рис. 2.6. Структура адресної пам'яті з довільним доступом:
РгА – k -бітний реєстр адреси пам'яті; **БПП** – блок підсилювачів прочитування; **ЗМ** – запам'ятовуючий масив; **БАВ** – блок адресної вибірки; **БУП** – блок управління пам'яттю; **БПЗ** – блок підсилювачів запису; **РгД** – реєстр даних.

При записі, крім виконання вказаної вище загальної части- ни циклу звернення, проводиться прийом слова, яке записуєть- ся, із вхідної шини даних (ШДвх) і РгД. Потім у вибрану БАВ комірку записується слово з РгД.

Блок управління БУП генерує необхідні послідовності сигна- лів і ініціює роботу окремих вузлів пам'яті.

Асоціативна пам'ять. У пам'яті такого типу пошук потрібної інформації проводиться не за адресою, а за її змістом (за асоці- ативною ознакою). При цьому пошук за асоціативною ознакою

(або послідовно за окремими розрядами цієї ознаки) відбувається паралельно в часі для всіх комірок запам'ятовуючого масиву. У багатьох випадках асоціативний пошук дозволяє істотно спростити і прискорити оброблення даних. Це досягається за рахунок того, що в пам'яті цього типу операція читання інформації суміщена з виконанням певних логічних операцій.

Типова структура асоціативної пам'яті представлена на рис. 2.7.

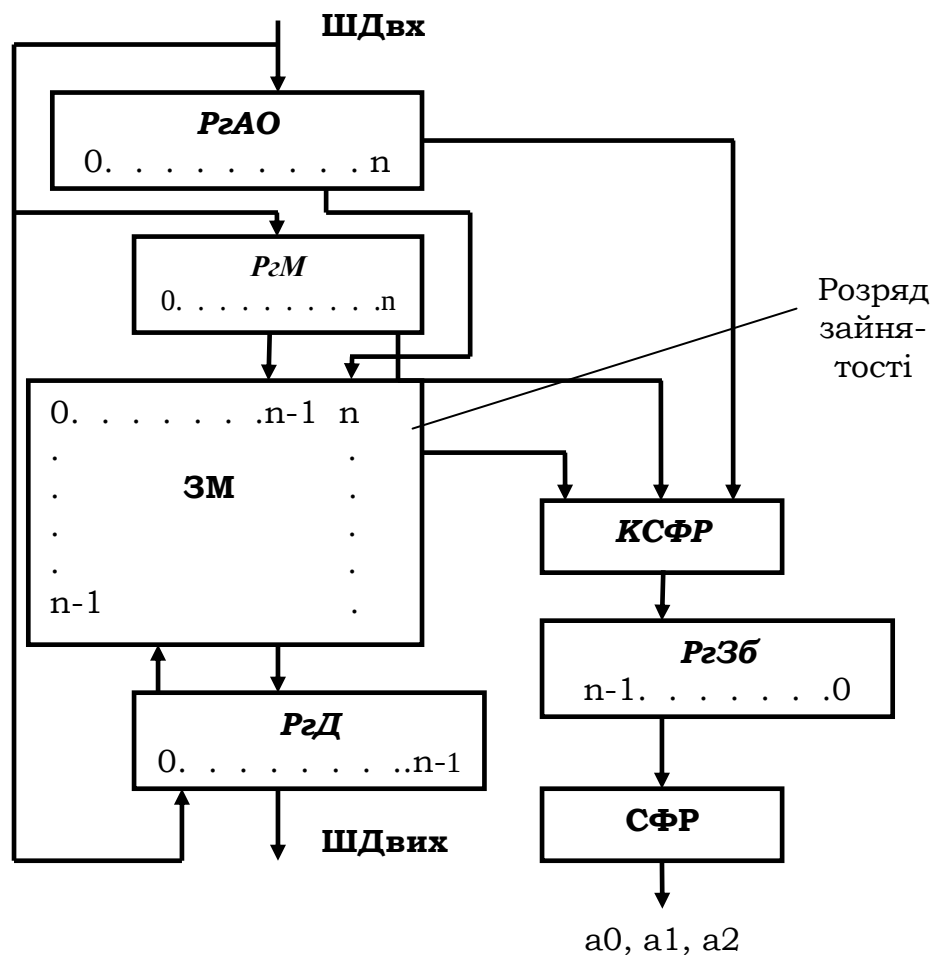


Рис. 2.7. Структура асоціативної пам'яті:

P_2AO – реєстр асоціативної ознаки; P_2M – реєстр маски;
 ZM – запам'ятовуючий масив; $P_2Д$ – реєстр даних; КС – комбінаційна
схема; $P_2Зб$ – реєстр збігу; СФР – схема формування результату;
ШДвх – вхідна шина даних; ШДвих – вихідна шина даних

Запам'ятовуючий масив містить $n(n+1)$ розрядних комірок. Для вказівки на зайнятість комірки використовується службовий n -й розряд (0 – комірка вільна, 1 – у комірці записане слово).

По вхідній шині даних (ШДвх) у реєстр асоціативної ознаки (P_2AO) в розряди $0 \dots n-1$ поступає n -розрядний асоціативний запит, а в реєстр маски (P_2M) – код маски пошуку, при цьому n -й розряд P_2M встановлюється в 0. Асоціативний пошук проводиться лише для сукупності розрядів P_2AO , яким відповідають одиниці в P_2M (незамасковані розряди P_2AO). Для слів, у

яких цифри в розрядах співпали з незамаскованими розрядами $RgAO$, комбінаційна схема (КС) встановлює в 1 відповідні розряди регістра збігу ($Rg3b$) і в 0 решту розрядів.

Комбінаційна схема формування результату формує зі слова, що утворилося в $Rg3b$, сигнали $a0$, $a1$, $a2$, які відповідають випадкам відсутності слів у ЗМ, що задовольняють асоціативній ознаці, або наявності одного (або більше) такого слова.

Формування вмісту $Rg3b$ і сигналів $a0$, $a1$, $a2$ за вмістом $RgAO$, RgM і ЗМ називається **операцією контролю асоціації**. Ця операція є складовою частиною операцій читання і запису, хоча вона має і самостійне значення.

При читанні спочатку проводиться контроль асоціації за асоціативною ознакою в $RgAO$. Потім при $a0 = 1$ читання відміняється через відсутність відповідної інформації, при $a1 = 1$ читається в RgD знайдене слово, при $a2 = 1$ в RgD читається слово з комірки, що має найменший номер серед комірок, позначених одиницями в $Rg3b$. З RgD знайдене слово видається на ШДвих.

При записі спочатку відшукується вільна комірка. Для цього виконується операція контролю асоціації при $RgAO = 111...10$ і $RgM = 00...01$, при цьому вільні комірки відмічаються одиницями в $Rg3b$. Для запису вибирається вільна комірка з найменшим номером. У неї записується слово, що поступило з ШДвх в RgD .

За допомогою операції контролю асоціації можна, не прочитуючи слів з пам'яті, визначити за вмістом $Rg3b$, скільки в пам'яті слів, що задовольняють асоціативній ознаці, наприклад, реалізувати запити типу «Скільки студентів у групі мають відмінну оцінку з даної дисципліни». При використанні відповідних комбінаційних схем в асоціативній пам'яті можуть виконуватися досить складні логічні операції, такі як пошук більшого (меншого) числа, пошук слів, пошук максимального (мінімального) числа та ін. Асоціативна пам'ять застосовується, наприклад, у пристроях динамічного розподілу ОП.

Відзначимо, що для асоціативної пам'яті необхідні запам'ятовуючі елементи, що допускають прочитування без руйнування записаної в них інформації. Це пов'язано з тим, що при асоціативному пошуку прочитування проводиться по всьому ЗМ для всіх незамаскованих розрядів і зберігати тимчасово зруйновану прочитуванням інформацію нема де.

Стекова пам'ять, так само як асоціативна, є безадресною. Стекову пам'ять можна розглядати як набір комірок, які утворюють одномірний масив (див. рис. 2.8). Запис нового слова здійснюється у верхню комірку (з номером 0). При цьому всі

раніше записані слова (як і слово в комірці з номером 0) зсуваються униз, у сусідні комірки з номерами на 1 більшими від власних. Читання можливе лише з верхньої (нульової) комірки пам'яті. Якщо читання суміщене з видаленням, то всі слова в пам'яті, що залишаються, зсуваються вгору, у сусідні комірки з більшими номерами.

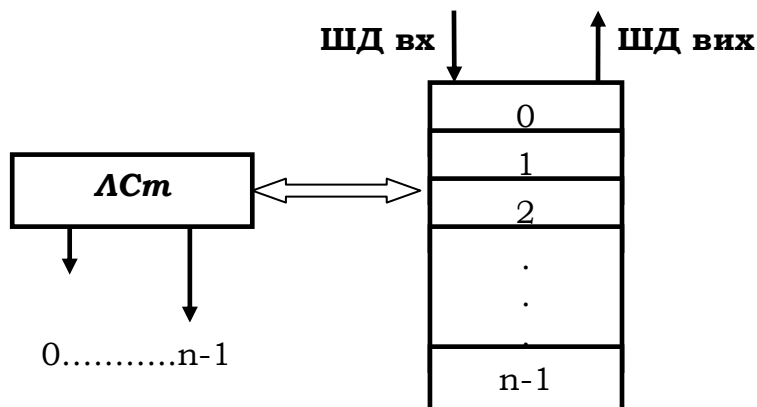


Рис. 2.8. Стекова пам'ять:
 ЛСт – лічильник стека; ШДвх – вхідна шина даних;
 ШДвих – вихідна шина даних

У стековій пам'яті порядок читання слів відповідає правилу «останній прийшов – першим вийшов». У ряді пристроїв такого типу передбачена операція читання без видалення та зсуву слова в пам'яті. Іноді стекова пам'ять забезпечується лічильником стека ЛСт, який показує кількість занесених у пам'ять слів. Сигнал ЛСт = 0 відповідає порожньому стеку, сигнал ЛСт = n-1 – заповненому стеку.

Зазвичай стекову пам'ять організовують, використовуючи адресну пам'ять. У цьому випадку лічильник стека ЛСт, як правило, відсутній, оскільки кількість слів у пам'яті можна виявити за показником стека. Найбільше поширення стекова пам'ять знаходить при обробленні вкладених структур даних, при виконанні безадресних команд і переривань.

Питання для самоперевірки

1. У чому полягають основні принципи роботи цифрових ЕОМ?
2. Наведіть основні пристрої, із яких складається класична ЕОМ.
3. Наведіть класифікацію ЕОМ.
4. Які Ви знаєте характеристики ЕОМ?
5. У чому полягає суть програмного управління ЕОМ?
6. Наведіть основні особливості магістрально-модульного типу архітектури ЕОМ?

7. Визначте суть понять: «слово даних», «адреса», «магістраль» (або «шина»), «пам'ять».
8. Що таке мікропроцесор?
9. Що таке вектор стану програми?
10. Поясніть особливості асоціативного типу пам'яті.
11. Розкрийте особливості адресного типу пам'яті.
12. Як здійснюються процеси запису/читання в пам'ять типу «стек»?

РОЗДІЛ 3

МІКРОПРОЦЕСОРНА СИСТЕМА ТА ЇЇ ФУНКЦІОНУВАННЯ

Архітектура МПС або цифрового пристрою поєднує апаратні, мікропрограмні і програмні засоби та визначає особливості конструювання, програмування та використання цифрового пристрою.

3.1. КЛАСИФІКАЦІЯ МІКРОПРОЦЕСОРІВ

Безупинне вдосконалення технології і схемотехніки великих інтегральних мікросхем (ВІС) обумовило створення і розвиток одних із найбільш універсальних ВІС – мікропроцесорів. До сьогодення часу вітчизняною промисловістю і закордонними фірмами випускається велика кількість мікропроцесорних ВІС, що відрізняються швидкодією, потужністю, розрядністю й іншими характеристиками, а також функціональним набором схем у комплекті. Функціональний склад комплектів мікропроцесорних ВІС і їх основні характеристики визначалися на основі аналізу різних варіантів застосування мікропроцесорів і оптимізації алгоритмів керування з урахуванням вимог до продуктивності й обсягу пам'яті.

Високий рівень технічних характеристик мікропроцесорних ВІС і порівняно низька вартість у розрахунку на одиницю інформації стимулювали швидке впровадження мікропроцесорів у побутову, промислову і спеціальну радіоелектронну апаратуру.

Мікропроцесори мають складну логічну структуру і містять велику кількість електронних елементів з безліччю розгалужених зв'язків між ними. Для підвищення продуктивності мікропроцесора необхідно використовувати всі його апаратні ресурси.

Необхідність виконання складних функцій керування призвела до створення **мікроконтролерів** – керуючих пристроїв, виконаних на одному чи кількох кристалах. Мікроконтролери виконують функції логічного аналізу і керування. За рахунок виключення можливості виконання арифметичних операцій зменшилась апаратна складність мікроконтролерів (що призвело до здешевлення цього пристрою).

Класифікація мікропроцесорів

1. За функціональним призначенням розрізняють універсальні і спеціалізовані МП (див. рис. 3.1).

Універсальні МП мають алгоритмічно універсальний набір команд, за допомогою якого можна здійснювати перетворення інформації відповідно до будь-якого заданого алгоритму. Продуктивність (швидкодія) таких процесорів практично не залежить від специфіки розв'язуваних задач.

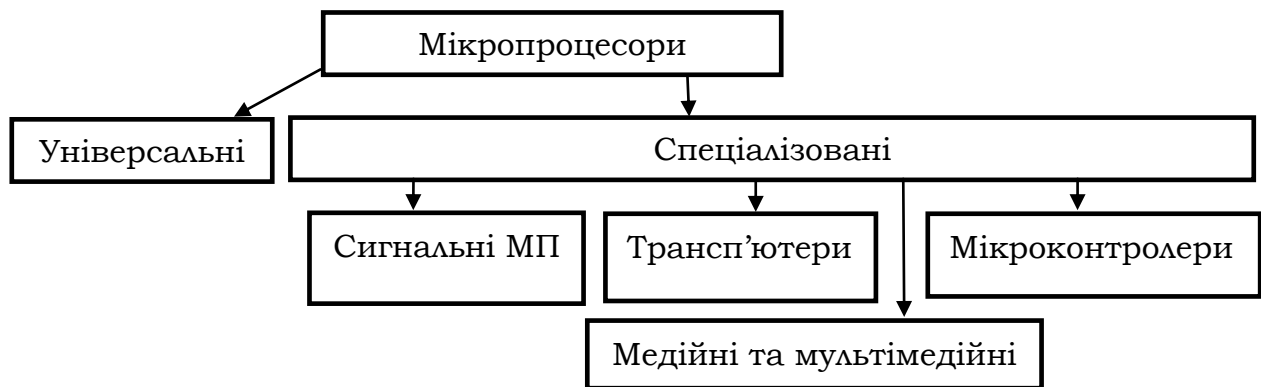


Рис. 3.1. Класифікація МП за функціональним призначенням

Спеціалізовані МП призначені для вирішення обмеженого і строго визначеного кола задач, іноді навіть для вирішення однієї конкретної задачі. До спеціалізованих МП належать: сигнальні; медійні та мультимедійні; трансп'ютери; мікроконтролери.

Сигнальні процесори (процесори цифрових сигналів) призначені для цифрового оброблення сигналів у реальному масштабі часу (наприклад, фільтрація сигналів, обчислення згортки та кореляційної функції, підсилення, обмеження та трансформація сигналу, пряме та обернене перетворення Фур'є).

Медійні та мультимедійні процесори призначені для оброблення аудіосигналів, графічної інформації, відеозображень, а також для розв'язування ряду задач у мультимедіакомп'ютерах, іграшкових приставках, побутовій техніці.

Трансп'ютери призначені для масових паралельних обчислень і роботи в мультипроцесорних системах. Для них характерним є наявність внутрішньої пам'яті та вбудованого міжпроцесорного інтерфейсу, тобто каналів зв'язку з іншими МП.

Серед спеціалізованих МП також можна виділити мікроконтролери – МП, призначені для вирішення задач керування якими-небудь процесами або пристроями. На допомогу центральному процесорові в комп'ютер іноді вводять співпроцесори, орієнтовані на ефективне виконання яких-небудь специфічних функцій. Раніше були досить поширені математичні співпроцесори, що обробляють числові дані у форматі з рухомою комою; графічні співпроцесори, що виконують геометричні побудови й оброблення графічних зображень; співпроцесори введення/виведення, що розвантажують центральний процесор від нескладних, але численних операцій взаємодії із зовнішніми пристроями. Можливі й інші співпроцесори, однак усі вони несамоостійні – виконання основного обчислювального процесу здійснюється центральним процесором, що відповідно до програми видає «завдання» співпроцесорам на виконання їхніх допоміжних функцій.

2. За числом ВІС, які використовуються для побудови функціонально повного МП, розрізняють однокристалні, багатокристалні і багатокристалні секційні МП.

Однокристалні МП реалізуються у вигляді однієї ВІС. Логічна структура багатокристалного МП розбивається на складні функціональні вузли, кожен з яких реалізується у вигляді окремої ВІС.

Однокристалні МП можуть бути одноядерними та багатоядерними. Багатоядерний процесор має фактично кілька обчислювальних пристроїв, розташованих на одному кристалі, які можуть працювати паралельно та мають загальні контакти для взаємодії з шиною.

Багатокристалні секційні (розрядно-модульні) МП використовуються для побудови багаторозрядних МП на основі паралельно включених мікропроцесорних секцій. Мікропроцесорна секція являє собою ВІС для оброблення декількох розрядів даних (від 2 до 16), що може використовуватися як самостійно, так і як модуль для побудови МП, що обробляє більш довгі кодові слова.

3. За розрядністю слів даних, які може обробляти МП, розрізняють МП з фіксованою або змінною розрядністю слів.

Процесорами зі змінною розрядністю слів даних є тільки багатокристалні секційні МП, усі інші МП обробляють слова фіксованої розрядності.

4. За способом керування обчислювальним процесом розрізняють МП із мікропрограмним або із схемним (апаратним) керуванням.

Мікропроцесори зі схемним керуванням мають фіксований набір команд, розроблений фірмою-виробником, який не може змінювати користувач. У МП з мікропрограмним керуванням систему команд розробляють при проектуванні конкретного мікропроцесорного комплекту на базі набору найпростіших мікрокоманд з урахуванням класу задач, для яких призначений пристрій.

Мікропрограмні багатокристалні МП забезпечують велику гнучкість у досягненні потрібних користувачу характеристик, у першу чергу потрібної швидкодії мікропроцесорного пристрою або мікропроцесорної системи, надаючи користувачу можливість задавати спеціалізовану систему команд, орієнтовану на певне використання. Проте при цьому проектувальник повинен розробляти мікропрограми, які реалізують ці команди, і записати їх в управляючу пам'ять МП.

За характером роботи в часі МП поділяють на синхронні і асинхронні.

Синхронні МП – це МП, у яких початок і кінець виконання операцій задається пристроєм управління (ПУ).

Асинхронні МП дозволяють початок виконання кожної наступної операції визначити за сигналом фактичного закінчення виконання попередньої операції.

За організацією структури МПС розрізняють:

- одномагістральні МПС;
- багатоманістральні МПС.

В одномагістральних МПС усі пристрої мають однаковий інтерфейс і підключені до єдиної системної маністралі, по якій передаються коди даних, коди адрес і управляючих сигналів. У багатоманістральних МПС пристрої групами підключаються до своєї маністралі.

За кількістю виконуваних програм розрізняють однопрограми і багатопрограми МП. В однопрограми МП перехід і виконання наступної програми відбувається після завершення поточної програми. Багатопрограми МП дозволяє виконувати одночасно кілька програм.

3.2. ТИПОВА СТРУКТУРА МІКРОПРОЦЕСОРНОГО ПРИСТРОЮ

Виконання того чи іншого алгоритму можливе за наявності мікропроцесора та пристроїв, у яких зберігається програма. Відомо, що програма – це сукупність команд (правил), що виконуються в послідовності, заданій алгоритмом. Команди вибираються з пам'яті в послідовності, що задається процесором. Процесор визначає адреси елементів пам'яті, у яких зберігаються необхідні дані. Дані передаються в процесор, де перетворюються у відповідності з командами, і результати операції передаються знову в пам'ять.

Будь-яка МПС працює разом із рядом зовнішніх пристроїв, одержуючи від них необхідну інформацію та передаючи іншу. Для зв'язку із зовнішнім пристроєм існує інтерфейс (англ. *interface*). Цим терміном позначається весь комплекс пристроїв, правил та технічних засобів, що регламентують та забезпечують обмін інформацією між мікропроцесором (включаючи пам'ять) та зовнішнім пристроєм. Головними в інтерфейсі є шини, або, як їх ще часто називають, маністралі. Маністраль – це сукупність провідників, для яких строго нормовані логічні рівні «0» та «1». Потужність сигналів на шині має бути достатньою для живлення необхідної кількості приєднаних до них пристроїв. Для забезпечення цієї потужності використовуються спеціальні мікросхеми – шинні підсилювачі.

За призначенням шини поділяються на три типи:

- 1) адресні;

- 2) даних;
- 3) управління.

Але реально як в мікропроцесорній техніці, так і в комп'ютерній часто дві шини суміщають шляхом мультиплексування, що дещо знижує їх швидкодію, але набагато зменшує кількість виводів мікросхем.

Узагальнена логічна структура МПС наведена на рис. 3.2.

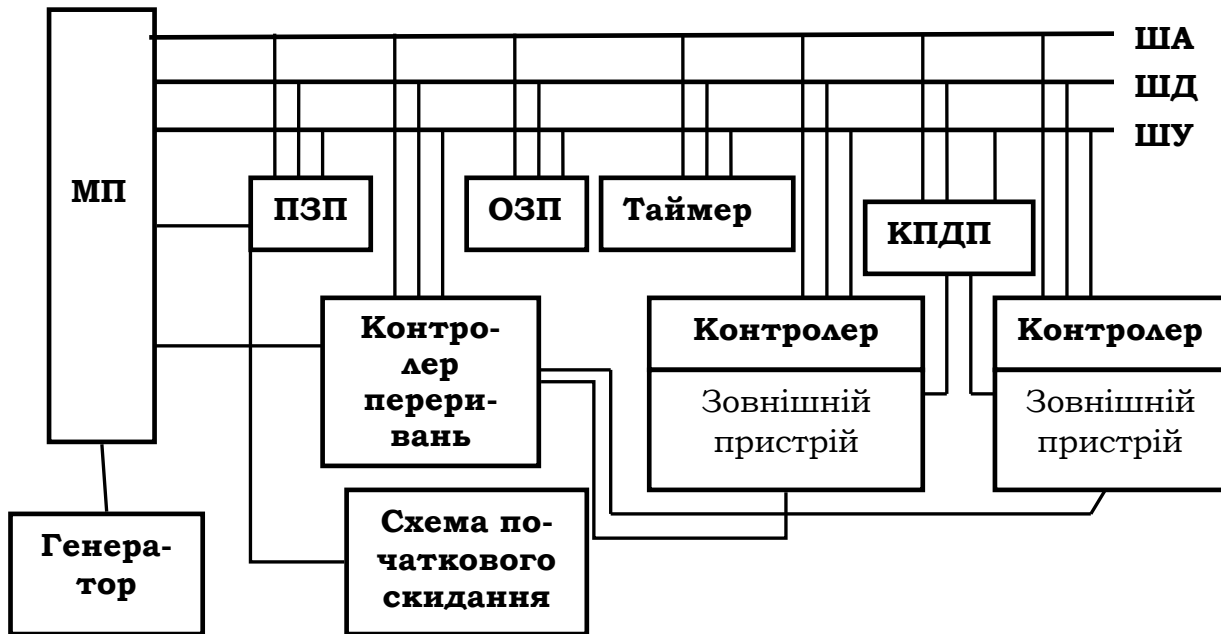


Рис. 3.2. Узагальнена логічна структура МПС:

МП – мікропроцесор, ША – шина адреси, ШД – шина даних,
 ШУ – шина управління, ОЗП – оперативний запам'ятовуючий пристрій,
 ПЗП – постійний запам'ятовуючий пристрій,
 КПДП – контролер прямого доступу до пам'яті

Основою будь-якої МПС є мікропроцесор. МП здійснює цифрове оброблення даних, виконання команд програми, що зберігається в пам'яті, за допомогою шин адреси, даних і управління.

Для внутрішнього опису функціонування МП необхідно охарактеризувати:

- формат даних, що обробляються, і команд;
- способи адресації даних;
- кількість внутрішніх регістрів загального призначення;
- організацію і адресацію стека;
- адресний простір та об'єм пам'яті.

У постійному запам'ятовуючому пристрої (ПЗП) зберігаються системні програми, необхідні для управління процесом оброблення даних.

В оперативному запам'ятовуючому пристрої (ОЗП) зберігаються прикладні програми, дані і результати обчислень.

Робота МП синхронізується тактовими сигналами *clk*, що поступають на його входи від генератора тактової частоти (синхронізації).

Схема початкового скидання виробляє сигнал *reset* (скидання) мікропроцесора на основі аналізу напруг на виході блока живлення (включення живлення) або при примусовій зупинці роботи МПС із клавіатури.

Контролери – пристрої, що забезпечують обмін даними, які надходять із різних зовнішніх пристроїв, між мікропроцесором і ОП. Контролер переривань – це контролер, який аналізує сигнали (запити на переривання), що надходять до МПС від зовнішніх пристроїв (зовнішні переривання) або мікропроцесора (внутрішні переривання) та організує відгук (або реакцію) МПС на переривання.

Контролер прямого доступу до пам'яті (КПДП) призначений для обміну даними між ЗП та ОП без участі мікропроцесора.

Залежно від конструктивних особливостей мікропроцесорного пристрою до його складу можуть входити інші складові, наприклад:

- шинний контролер, призначений для сполучення пристроїв із системною шиною по паралельному інтерфейсу;

- контролер послідовного інтерфейсу, призначений для побудови багатопроцесорних систем або для сполучення джерел і приймачів сигналів з метою зменшення навантаження на системний інтерфейс;

- спеціалізований процесор арифметичного оброблення сигналів (співпроцесор);

- ПЗП команд і констант;

- ОЗП операндів.

Для забезпечення роботи МПС до її системного інтерфейсу можна підключати пристрої для спеціалізованого оброблення арифметичних алгоритмів (таких як швидке перетворення Фур'є, десяткова корекція) і пристрої оброблення аналогових сигналів.

Як пристрої для оброблення аналогових сигналів використовують аналого-цифрові (АЦП) і цифро-аналогові (ЦАП) перетворювачі, що забезпечують безпосереднє сполучення цифрового пристрою з аналоговими сигналами від датчиків і приймачів.

3.3. СПОСОБИ ПЕРЕДАННЯ ЦИФРОВОЇ ІНФОРМАЦІЇ В МІКРОПРОЦЕСОРНІЙ СИСТЕМІ

Існує три способи передавання інформації:

- 1) програмно кероване передавання даних, що ініціюється процесором;

- 2) програмно кероване передання даних, що ініціюється запитом на переривання від зовнішнього пристрою;
- 3) прямий доступ до пам'яті (ПДП).

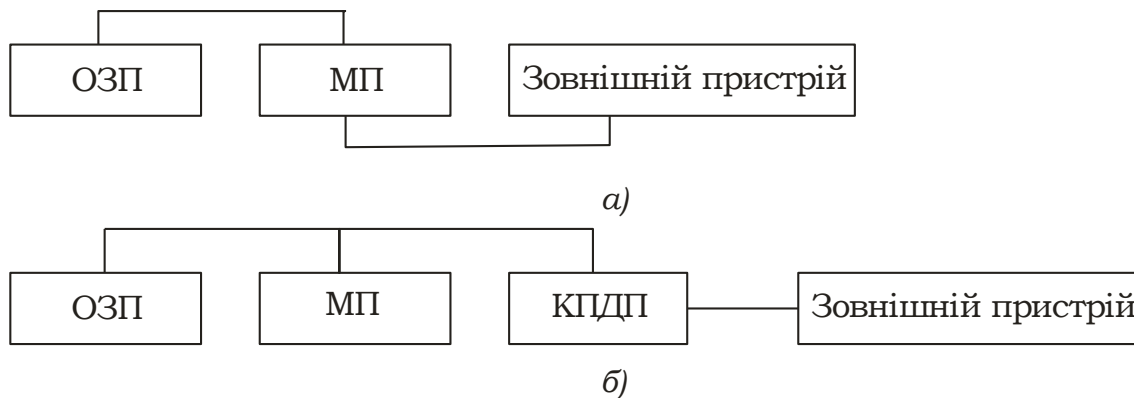


Рис. 3.3. Способи передання цифрової інформації:

а – програмно кероване передання даних;

б – прямий доступ до пам'яті

При програмно керованому переданні (див. рис. 3.3 а), передання слів даних відбувається через регістри процесора: слово даних через контролер зовнішнього пристрою надходить до внутрішнього регістра мікропроцесора (акумулятора) і залежно від потреб подальшого оброблення надходить до ОЗП.

При програмно керованому переданні даних МП на час цієї операції відволікається від виконання основної програми, що веде до зниження продуктивності МПС. Крім того, швидкість передання даних через МП може виявитися недостатньою для роботи з високошвидкісними ЗП.

Відрізняються перший і другий спосіб передання даних тільки ініціатором: при першому способі – це процесор (наприклад, при виконанні команди програми, яка потребує даних із зовнішнього пристрою), при другому – це зовнішній пристрій, який надсилає сигнал «Переривання».

Прямим доступом до пам'яті (див. рис. 3.3 б) називається обмін даними, що забезпечує автономне передання даних між ОЗП і ЗП (без участі мікропроцесора).

Прямий доступ до пам'яті збільшує граничну швидкість введення/виведення інформації і загальну продуктивність МПС, робить її більш пристосованою для роботи в системі реального часу.

Прямим доступом до пам'яті управляє КПДП, що виконує наступні функції:

- управління переданням даних між ОЗП і ЗП, що ініціюється процесором або ЗП;
- визначення розміру блока даних, що передаються, та області пам'яті, куди записуються дані;

- визначення адреси комірок ОЗП, що беруть участь у передаванні;
- підрахунок кількості байт, що передані через інтерфейс, та визначення моменту завершення операції введення/виведення.

3.4. ФУНКЦІОНУВАННЯ МІКРОПРОЦЕСОРНОЇ СИСТЕМИ

Функціонування МПС – це процес, який складається з виконання команд програми та обміну даними.

Процес функціонування МПС зводиться до наступної послідовності дій:

1) отримання даних від різних зовнішніх пристроїв (з клавіатури, від дисплеїв, з каналів зв'язку, від різного типу зовнішніх запам'ятовуючих пристроїв (дискети, флеш-карти, вінчестери));

2) оброблення даних;

3) видання результатів оброблення на зовнішні пристрої.

При цьому дані від ЗП, що підлягають обробленню, можуть надходити і в процесі їх оброблення.

Оперативна пам'ять зберігає і видає на запит команди програми, яку виконує мікропроцесор, різні дані (початкові, проміжні та кінцеві результати оброблення).

Мікропроцесор видає на шину адреси номер (адресу) комірки ОП, у якій записана чергова команда, і по шині управління в ОП надходять сигнали, що забезпечують читання вмісту цієї комірки. Чергова команда через шину даних передається в мікропроцесор. Тут команда розшифровується.

Якщо дані, які необхідні для виконання цієї команди, знаходяться в регістрах мікропроцесора, то мікропроцесор приступає до виконання вказаної в команді операції.

Якщо при розшифровці команди з'ясується, що дані, які беруть участь в операції, знаходяться в ОП, то мікропроцесор виставляє на шину адреси адресу комірки, де зберігаються ці дані; після видання даних з ОП мікропроцесор приймає їх через шину даних, потім виконується операція над даними.

Після завершення поточної команди на шину адреси видається адреса наступної команди, і описаний процес повторюється.

Обмін даними із ЗП може здійснюється наступним чином. Мікропроцесор, виконуючи команду введення, подає на контролер відповідні керівні сигнали; дані із ЗП приймаються в регістр контролера, потім вони контролером видаються на шину даних. Потім ці дані з шини даних приймаються в мікропроцесор. Далі в процесі виконання відповідної команди вони передаються в ОП.

Аналогічно відбувається обмін даними у зворотному напрямі – від ОП до ЗП. За відповідною командою програми здійснюється прийом з ОП у мікропроцесор даних, що підлягають переданню на ЗП, після чого за однією з наступних команд ці дані видаються на шину даних і через контролер обміну передаються на ЗП.

Описаний процес обміну припускає, що моменти обміну даними відомі наперед уже на етапі програмування, і в програмі передбачаються в певних місцях відповідні команди.

Моменти обміну можуть визначатися і самим ЗП. У таких випадках ЗП подає в мікропроцесор певні сигнали і переводить його в стан переривання. Перебуваючи в стані переривання мікропроцесор припиняє виконання основної програми і переходить до виконання команд іншої програми (перериваючої програми), зберігається в ОП. Після закінчення перериваючої програми мікропроцесор повертається до виконання основної програми.

Описані способи забезпечують низьку швидкість обміну, і застосовувати їх доцільно при обміні даними з низькошвидкісними ЗП.

При роботі з високошвидкісними ЗП (диск та ін.) використовується так званий метод прямого доступу до пам'яті (ПДП). У цьому режимі мікропроцесор відключається від шин адреси і даних, надаючи їх у розпорядження ЗП для безпосереднього обміну даними з ОП (без участі мікропроцесора). Обмін при цьому організовується спеціальним контролером ПДП.

У режимі ПДП ЗП обмінюється з ОП не бітами або байтами даних, а великими блоками даних. У контролер ПДП мікропроцесор заздалегідь поміщає інформацію, необхідну для управління обміном (адреса комірки ОП, куди записується або звідки прочитується перше слово обміну, кількість слів у блоці і ін.). У процесі обміну контролер ПДП видає на шину адреси адресу комірки ОП, після закінчення передання слова між ОП і ЗП через шину даних контролер ПДП збільшує на одиницю значення адреси, що видається на шину адреси. Після завершення передання заданої кількості слів контролер ПДП припиняє обмін, інформуючи про це мікропроцесор. Останній відновлює зв'язок з шинами адреси і даних та продовжує виконання програми.

3.5. ОРГАНІЗАЦІЯ ПРОСТОРІВ ПАМ'ЯТІ ТА ВВЕДЕННЯ/ВИВЕДЕННЯ

Організація просторів пам'яті в сучасних МПС доволі різноманітна і не піддається точній класифікації. Розглянемо найпростіші випадки.

Якщо в МПС для зберігання програм і даних використовується один простір пам'яті, говорять про *архітектуру Дж. фон*

Неймана. Програми і дані зберігаються в єдиному адресному просторі. Вміст комірки пам'яті прочитується та інтерпретується оператором оброблення, яким у найпростішому випадку виступає ЦП.

Проте майже всі однокристальні програмовані мікроконтролери виконані за іншою схемою, відомою як *архітектура Гарвардської лабораторії*, у якій пам'ять програм CSEG (*Code segment*) і пам'ять даних DSEG (*Data segment*) розділені і мають свої власні адресні простори і способи доступу до них. Таке розділення дозволило розмістити компактно кодований набір машинних команд, економно використовувати пам'ять програм і було застосоване при розробленні однокристальних мікроконтролерів перших типів, що мають всього лише 1–2 Кб управляючої пам'яті, розташованої на кристалі.

Подальше вдосконалення архітектури обох типів полягало у виділенні спеціального простору даних невеликого об'єму, який сьогодні відомий як набір програмно-доступних регістрів RSEG (*Register Segment*). На відміну від CSEG чи DSEG, регістри RSEG розташовуються усередині ЦП у безпосередній близькості від його арифметично-логічного пристрою (АЛП), що забезпечує швидкий фізичний доступ до інформації, котра зберігається в них. У деякі інтервали часу програма найбільш інтенсивно працює лише з невеликим об'ємом даних. Для тимчасового зберігання цих даних і призначена регістрова область – набір програмно-досяжних регістрів.

Область RSEG може бути як повністю ізольована від простору даних DSEG, так і частково перетинатися з ним, що дає можливість розглядати окремі регістри МП як звичайні елементи пам'яті даних. Така організація є доцільною, якщо в МПС підтримується швидкий доступ до всієї або хоча б деякої частини пам'яті даних DSEG.

Система введення/виведення (ВВ) у найпростішому випадку представляє набір буферних схем і регістрів (портів), через які здійснюється зв'язок із зовнішніми і внутрішніми апаратними засобами МПС. Система ВВ звичайно використовує єдиний механізм адресації портів, розміщених у спеціальному просторі ВВ мікросистеми IOSEG (*Input Output Segment*), логічно ізольованому від інших просторів даних, що адресуються, – ізольований ВВ. До МС з ізольованим ВВ відносяться системи на базі МП КР580ВМ80, К1810ВМ86 та ін., що мають спеціальні набори команд ВВ.

У деяких системах логічно ізольований простір ВВ може бути відсутній. У цьому випадку в просторі пам'яті даних DSEG виділяються області, у яких і розміщуються порти, – це суміщений

ВВ. Організація доступу до портів у таких МПС нічим не відрізняється від процесу запису/читання даних у пам'ять. Суміщений ВВ використовується в МПС на базі МП серії K1801.

На відміну від RSEG, пам'ять програм CSEG і даних DSEG, а також область ВВ IOSEG організовані простіше. У ряді випадків пам'ять МПС з погляду програміста є лінійно впорядкованим набором n -розрядних комірок із довільним доступом – лінійна пам'ять. Кожному набору комірок відповідає число, що називається її адресою. Усі адреси займають цілочисельний діапазон від 0 до $2^m - 1$, котрий утворює адресний простір пам'яті. Розрядність адреси m звичайно дорівнює 16, 18, 20, 24 або 32. У тих випадках, коли найменша одиниця пам'яті – байт ($n = 8$), пам'ять має байтову організацію.

Одним із прикладів МПС із пам'яттю лінійної організації байтового типу є 8-розрядна система на базі МП KP580BM80. Команди цього МП виконують звернення до простору пам'яті місткістю $2^{16} = 64$ Кб. Надалі пам'ять МПС представлятиметься таким чином, що комірки із старшими адресами розташовуватимуться нижче, ніж із молодшими. Нумерація окремих розрядів у комірці здійснюється з правого краю, починаючи з нуля. При цьому розряд із нульовим номером є молодшим.

За необхідності *програмні об'єкти*, команди і *операнди* (дані в командах), що зберігаються в пам'яті, можуть розташовуватися в сусідніх комірках простору пам'яті. Адресою об'єкта звичайно служить якнайменша з адрес комірок, займаних ним. Операція звернення до пам'яті припускає читання або запис всього об'єкта як єдиного цілого. Наприклад, слова в пам'яті МПС на базі МП KP580BM80 зберігаються у двох сусідніх байтах. Старша частина слова займає байт із старшою адресою, а молодша – байт із молодшою адресою. При цьому адреса молодшого байта слугує адресою слова.

Пам'ять більшості 16- і 32-розрядних МПС також має байтову організацію. Так, нижній рівень логічного представлення пам'яті МП K1810BM86 місткістю 1 Мб аналогічний розглянутому вище. Проте в даному МП існує більш високий рівень організації пам'яті, на якому в основному і працює програміст.

Дуже часто організація пам'яті передбачає певні обмеження на можливе розташування багатобайтових об'єктів. Так, у МС на базі 16-розрядного МП K1810BM86, яка також має пам'ять із байтовим доступом місткістю 64 Кб, слова в пам'яті можуть знаходитися тільки за парними адресами. Тоді при доступі до слова значення молодшого розряду його адреси, що вказує на байт у слові, до уваги не береться, тобто така пам'ять має межу слів. Порядок розташування байтів усередині слова стандартний: спочатку молодший, потім старший байт слова.

Організацію такого вигляду мають не тільки простори пам'яті DREG і CSEG, але й область BB IOSEG.

3.6. КОМАНДНИЙ ЦИКЛ

Центральний процесор здійснює введення, оброблення і виведення даних відповідно до програми, що зберігається в CSEG.

Програма – це впорядкована послідовність команд і даних. Процес виконання програми полягає в послідовному виконанні команд, із яких складається програма.

Команда – це функціональна завершена елементарна дія, яка визначається типом даних, що використовуються, джерелом їх отримання, операцією над ними, приймачем розміщення результату, а також джерелом отримання наступної команди. Програміст розглядає команду як одну неподільну дію. На рівні фізичного обміну кожна команда є рядом типових циклів звернення до системної магістралі.

Машинне представлення команди в пам'яті МПС називається її **об'єктним кодом**. Об'єктний код команди складається з ряду нулів і одиниць. Проте людині більш зрозуміла інформація, представлена в символьній формі. Тому разом з об'єктним кодом кожній команді приписується її символічне позначення, або **мнемокод**, який використовується при написанні програми людиною з подальшим її перекодуванням в машинне уявлення. Звичайно існує взаємно-однозначна відповідність між мнемокодом і об'єктним кодом команди.

МПС функціонує синхронно з частотою тактових сигналів зовнішнього генератора. Залежно від типу мікропроцесорів використовується одно- або двофазна синхронізація. Незалежно від цього в мікропроцесорних системах використовуються більш тривалі інтервали часу, ніж тактовий інтервал зовнішнього генератора. Одним із таких інтервалів є машинний цикл – це інтервал, протягом якого мікропроцесор звертається до пам'яті або пристрою введення/виведення. Машинний цикл складає тільки частину циклу команди. На початку кожного машинного циклу на одному з виходів мікропроцесора з'являється сигнал синхронізації, він передається по лінії шини управління в пам'ять або пристрій введення/виведення і «сповіщає» про початок нового машинного циклу, у результаті чого досягається узгодження в часі зовнішніх пристроїв з роботою мікропроцесора.

Цикл команди – це інтервал часу, необхідний для вибірки з пам'яті команди та на її виконання. Він триває 1–5 машинних циклів. Їх конкретна кількість залежить від складності операції,

яка виконується в даній команді, і дорівнює числу звернень мікропроцесора до пам'яті. Тривалість виконання команди визначається кількістю тактів у циклі команди та тривалістю такту.

Протягом циклу команди, що ділиться на дві фази, робота мікропроцесора виконується в такій послідовності. Пристрій управління (ПУ) задає початок чергового циклу шляхом формування сигналу, по якому число, що знаходиться в лічильнику команд, відправляється в буферний регістр адреси і через нього направляється для дешифрації. Після приходу від мікропроцесора управляючого сигналу «Готовий» із елемента пам'яті, що знаходиться за вказаною адресою, зчитується слово команди, яке подається по шині даних у буферний регістр даних, а потім – у пристрій управління, де дешифрується. Ця послідовність операцій називається *фазою вибірки*. За нею настає *фаза виконання*, у якій пристрій управління формує послідовність сигналів, необхідних для виконання команди. За цей час число, що знаходиться в лічильнику команд, збільшується на 1 (якщо довжина команди є 1), і формується адреса наступної команди. Вона зберігається в лічильнику до приходу сигналу, який сповіщає про початок чергового циклу команди.

Окрім адреси елемента, у якому зберігається необхідний байт, від мікропроцесора до пам'яті надходить сигнал по шині управління, який визначає характер операції – запис або зчитування. Виконання вказаних операцій проходить протягом інтервалу часу, що називається *часом доступу*. По закінченні цього інтервалу в мікропроцесор подається сигнал готовності, який є сигналом початку прийому, або, відповідно, передання сигналів у пам'ять. До одержання сигналу готовності мікропроцесор знаходиться в стані очікування. Інтервал часу між імпульсами звернення до зовнішніх пристроїв та одержання від них відповіді називається *циклом очікування*.

Якщо, наприклад, цикл команди розглядати відповідно до команди введення даних, то перші два машинних цикли будуть відноситись до фази вибірки, а третій – до фази виконання команди. В усіх машинних циклах передається адреса, але в кожному циклі адреса належить своєму адресату. У першому – це адреса комірки, де зберігається код операції, у другому – адреса порту, з якого передається байт даних, у третьому – адреса акумулятора мікропроцесора, куди повинен надійти байт даних із порту.

У цілому робота МПС полягає в наступному. При включенні джерела живлення або натисненні клавіші скидання *reset* управління апаратно передається на стартову адресу пам'яті програм. Вибирається і виконується перша команда, за наслідками якої

управління передається іншій і т. д. При прийомі спеціальної команди HLT мікропроцесорна система припиняє свою роботу до наступного пуску.

3.7. ПРИНЦИПИ ОРГАНІЗАЦІЇ ТА КЛАСИФІКАЦІЯ АРИФМЕТИКО-ЛОГІЧНИХ ПРИСТРОЇВ

Арифметико-логічні пристрої (АЛП) слугують для виконання арифметичних і логічних перетворень над словами, званими в цьому випадку операндами. АЛП є основною частиною операційного блоку ЕОМ.

Операції, які виконує АЛП, можна розділити на наступні групи:

- операції двійкової арифметики для чисел із фіксованою комою;
- операції двійкової арифметики для чисел із рухомою комою;
- операції десяткової арифметики;
- операції індексної арифметики;
- операції спеціальної арифметики;
- операції над логічними кодами;
- операції над алфавітно-цифровими полями.

Сучасні ЕОМ загального призначення звичайно реалізують операції всіх наведених вище груп, а малі і мікроЕОМ часто не мають апаратури арифметики чисел із рухомою комою, десяткової арифметики і операцій над алфавітно-цифровими полями. У цьому випадку операції реалізуються спеціальними програмами.

До арифметичних операцій відносяться складання, віднімання, обчислення модулів («короткі операції»), множення і ділення («довгі операції»).

Групу логічних операцій складають операції диз'юнкція (логічне «або») і кон'юнкція (логічне «і») над багаторозрядними двійковими словами, а також операція порівняння кодів на рівність.

Спеціальні арифметичні операції включають нормалізацію, арифметичний зсув (зсовуються тільки цифрові розряди, а знаковий залишається на місці), логічний зсув (знаковий розряд зсовується разом із цифровими).

За способом дії над операндами АЛП діляться на послідовні і паралельні. У послідовних АЛП операнди представляються в послідовному коді, а операції проводяться послідовно в часі над їх окремими розрядами. У даний час АЛП цього типу ніде не застосовуються. У паралельних АЛП операнди представляються паралельним кодом і операції виконуються паралельно в часі над всіма розрядами операндів.

За способом представлення чисел розрізняють АЛП:

- 1) для чисел із фіксованою комою;
- 2) для чисел із рухомою комою;
- 3) для десяткових чисел.

За характером використання елементів і вузлів АЛП діляться на блокові і багатофункціональні. У блоковому АЛП операції над числами з фіксованою і рухомою комою, десятковими числами і алфавітно-цифровими полями виконуються в окремих блоках, при цьому підвищується швидкість роботи, оскільки блоки можуть паралельно виконувати відповідні операції, але значно збільшуються витрати на устаткування. У багатофункціональних АЛП операції для всіх форм представлення чисел виконуються одними й тими ж схемами, які комутуються певним чином залежно від необхідного режиму роботи.

3.8. ПРОЕКТУВАННЯ КОМАНД

Проектування системи команд впливає на структуру ЕОМ. Оптимальну систему команд іноді визначають як сукупність команд, що задовольняє вимогам проблемно-орієнтованих задач виключити надмірність апаратних і апаратно-програмних засобів для реалізації команд, які рідко використовуються. У різних програмах ЕОМ частота появи команд різна; наприклад, за даними фірми DEC у програмах для ЕОМ сімейства PDP-11 найбільш часто зустрічається команда передачі MOV(B), її частка складає приблизно 32 % усіх команд у типових програмах. Систему команд варто вибирати таким чином, щоб витрати на рідко використововувані команди були мінімальними.

При наявності статистичних даних можна розробити (вибрати) ЕОМ з ефективною системою команд. Одним із підходів до досягнення даної мети є розроблення команд довжиною в одне слово і кодування їх таким чином, щоб розряди таких коротких команд використовувати оптимально, що дозволить скоротити час реалізації програми і її довжину.

Іншим підходом до оптимізації системи команд є використання мікроінструкцій. У цьому випадку окремі біти або групи біт команди використовуються для кодування декількох елементарних операцій, що виконуються в одному командному циклі. Ці елементарні операції не вимагають звертання до пам'яті, а послідовність їхньої реалізації визначається апаратною логікою.

Скорочення часу виконання програм і ємності пам'яті досягається за рахунок збільшення складності логіки управління.

Важливою характеристикою команди є її формат, що визначає структурні елементи команди:

– код операції, що визначає дію, яку буде виконувати процесор;

- адресу комірки пам'яті, регістра процесора або зовнішнього пристрою;
- режим адресації;
- операнд (наприклад, константа) при використанні безпосередньої адресації;
- коди ознак, які аналізуються при виконанні команд умовного переходу.

Рівні формалізації

Найнижчим рівнем, який дозволяє описувати роботу цифрових пристроїв, – це рівні логічних станів їх входів та виходів – таблиці станів.

Наступним рівнем є спосіб описання – це мова значень вхідних та вихідних сигналів, що складають мову мікрокоманд. Сукупність адрес та керуючих сигналів називаються мікрокомандою.

Третій рівень формалізації опису роботи мікропроцесора – це мова команд, тобто однозначна послідовність мікрокоманд, що записується в пам'яті мікропроцесорів. Тобто команда – це слово, або набір слів, які дешифруються в послідовність мікрокоманд. Звідси витікає, що будь-який процесор має строго фіксований і обмежений набір команд, який є характерним для даного процесора. Будь-яка мікрокоманда характеризується своїм форматом. Під форматом мікрокоманди розуміється її довжина та призначення кожного біта або їх групи. Довжина мікрокоманди – це стандартна для даного процесора кількість біт у слові. Команди також мають свій фіксований формат. Залежно від команди вона може складатися з одного, двох або трьох слів.

Формат пам'яті мікропроцесорної системи також тісно пов'язаний із довжиною слова. Для зберігання таких команд відповідно використовується адресний простір пам'яті. Якщо, наприклад, команда складається з трьох слів, а адресний простір має послідовну адресацію, то для зберігання такої команди використовуються три послідовні адреси. Для того щоб таку команду вибрати з пам'яті, необхідно мати спеціальні засоби, щоб забезпечити її представлення як єдине ціле.

Усі можливі перетворення дискретної інформації можуть бути зведені до чотирьох основних видів:

- 1) передання інформації в просторі;
- 2) зберігання інформації;
- 3) логічні операції;
- 4) арифметичні операції.

ЕОМ, що є універсальним перетворювачем дискретної інформації, виконує всі вказані види перетворень.

Оброблення інформації в ЕОМ здійснюється автоматично шляхом програмного управління. Програма є алгоритмом

оброблення інформації, записаним у вигляді послідовності команд, які повинні бути виконані машиною для отримання рішення задачі.

Команда є кодом, що визначає операцію обчислювальної машини і дані, що беруть участь в операції. Команда містить також у явній або неявній формі інформацію про адресу, за якою поміщається результат операції, і про адресу наступної команди.

Процес виконання програми складається з окремих машинних операцій. У даному випадку під операцією розуміється перетворення інформації, виконуване машиною під впливом однієї команди.

Змістом машинної операції можуть бути запам'ятовування в пам'яті, передання, арифметичне або логічне перетворення машинних слів, а також деякі допоміжні процедури.

3.9. СТРУКТУРА ТА КЛАСИФІКАЦІЯ КОМАНД

Структура команд повністю залежить від структури мікропроцесора, але незалежно від типу процесора прийнято вважати, що однослівні команди повністю складаються з коду операції. Двослівні команди складаються з коду операції та однослівного операнда. Трислівні команди також складаються з двох частин: перша частина – код операції, а друга – адреса, або двослівний операнд.

Типи команд, що використовуються, тісно пов'язані з внутрішньою організацією та алгоритмом функціонування мікропрограмного автомата процесора та внутрішньою системою синхронізації.

Найважливішим структурним елементом формату будь-якої команди є код операції (КОП), що визначає дію, яка має бути виконана. Велика кількість КОП у процесора дуже важлива, тому що апаратна реалізація команд заощаджує пам'ять і час. Але при виборі ЕОМ необхідно концентрувати увагу на повноті операцій із конкретними типами даних, а не тільки на числі команд та доступних режимах адресації. Число біт, що відводиться під КОП, є функцією повного набору реалізованих команд.

При використанні фіксованого числа біт під КОП для кодування всіх m команд необхідно в поле КОП виділити 2^{m-1} двійкових розрядів. Однак з огляду на обмежену довжину слова малих ЕОМ, різне функціональне призначення команд, а також те, що не всі команди містять адресну частину для звертання до пам'яті і зовнішніх пристроїв, у малих ЕОМ для кодування команд широко використовується принцип кодування з перемінним числом біт КОП для різних груп команд.

У деяких командах необхідним є тільки один операнд, такі команди називаються однооперандними (або одноадресними) командами, на відміну від двооперандних (або двоадресних), у яких потрібні два операнда. При наявності двох операндів командою звичайно змінюється тільки один із них. Це пояснюється тим, що інформація береться тільки з одного учасника операції – джерела; другий учасник операції, вміст якого змінюється, називається приймачем.

Найбільш гнучка команда вимагає до чотирьох операндів. Наприклад, команда додавання може вказувати адреси учасників операції, що складаються одна з одною, адресу результату й адресу наступної команди. Якщо для задання адреси потрібно 16 біт, то чотирьохоперандна команда буде займати 8 байт пам'яті, не враховуючи код операції. Отже, вийде повільнодіюча ЕОМ із величезною пам'яттю. Тому в більшості малих ЕОМ у командах використовують не більше двох операндів. Це досягається наступними прийомами:

- адреса наступної команди вказується тільки в командах переходів; в інших випадках чергова команда вибирається з комірок пам'яті, що розташовані за черговою командою;

- використання комірки, у якій знаходиться один з операндів, для запам'ятовування результату (наприклад, сума запам'ятовується в комірці першого операнда).

Локалізацію і звертання до операндів забезпечують режими адресації. При введенні декількох режимів адресації для одного коду операції необхідно відвести в команді біти, що вказують режими адресації для кожного операнда. Наприклад, якщо передбачено вісім режимів адресації, то для завдання кожного з них потрібно три біти.

Майже у всіх форматах команд перші біти відводяться коду операції, але далі формати команд різних ЕОМ сильно відрізняються один від одного. Наступні біти повинні визначати операнди або їхні адреси, і тому вони використовуються для комбінації режимів, адрес реєстрів, адрес пам'яті, відносних адрес і безпосередніх операндів. Звичайно довжина команди варіюється від 1 до 3 і навіть 6 байт.

За характером виконуваних операцій розрізняють наступні основні групи команд:

- 1) команди арифметичних операцій для чисел із фіксованою і рухомою комами;
- 2) команди десятикової арифметики;
- 3) команди логічних операцій;
- 4) команди передавання кодів;
- 5) команди введення-виведення;

- 6) команди передання управління;
- 7) команди завдання режиму роботи машини.

У команді, як правило, містяться не самі операнди, а інформація про адреси елементів пам'яті або регістри, у яких вони знаходяться.

Таким чином, структура команди визначається складом, призначенням і розташуванням полів у команді. Форматом команди називають її структуру з розміткою номерів розрядів (бітів), що визначають межі окремих полів команди, або з вказівкою числа біт в певних полях.

3.10. АДРЕСАЦІЯ В КОМАНДАХ МІКРОПРОЦЕСОРА

Слід розрізняти поняття *адресний код* (АК) в команді і *виконавська адреса* (ВА). Адресний код – це інформація про адресу операнда, що міститься в команді. Виконавська адреса – це номер елементу пам'яті, до якого проводиться фактичне звернення. У сучасних ЕОМ адресний код, як правило, не співпадає з виконавською адресою.

Вибір способів адресації, формування виконавської адреси і перетворення адрес є найважливішими питаннями розробки ЕОМ. Розглянемо способи адресації, які використовуються в сучасних ЕОМ.

Операнд, що мається на увазі. У команді не міститься явної вказівки на адресу операнда; операнд мається на увазі і фактично задається кодом операції команди. Даний спосіб використовується не часто, проте є кілька важливих випадків його застосування. Як приклад можна навести команди підрахунку, у яких до деякого числа (наприклад, вмісту лічильника) додається фіксований приріст, частіше одиниця молодшою розряду. Один з операндів – число в лічильнику – зазвичай адресується явним способом, другий операнд – приріст – не адресується, у пам'яті машини не міститься і є таким, що мається на увазі.

Адреса, що мається на увазі. У команді не міститься явної вказівки на адресу операнда, що бере участь в операції, або адреси, за якою поміщається результат операції, але ця адреса мається на увазі. Наприклад, команда може містити адреси обох операндів, що беруть участь в операції, при цьому мається на увазі, що результат операції поміщається за адресою одного з операндів, або команда указує тільки адресу одного операнда, а адреса другого, яким є вміст спеціального регістра (званого регістром результату або акумулятором), мається на увазі.

Безпосередня адресація. У команді міститься не адреса операнда, а безпосередньо сам операнд. При такій адресації не вимагається звернення до пам'яті для вибірки операнда і осередку для його зберігання. Це сприяє зменшенню часу

виконання програми і займаного нею об'єму пам'яті. Безпосередня адресація зручна для зберігання різного роду констант, проте слід мати на увазі, що при цьому способі адресації довжина операнда коротша за код команди, оскільки частина розрядів команди зайнята під код операції.

Пряма адресація. Виконавська адреса співпадає з адресною частиною команди. Цей спосіб адресації був загальноприйнятим в обчислювальних машинах і продовжує застосовуватися в даний час у комбінації з іншими способами.

У вказаній формі безпосередня адресація реалізується в ЕОМ з порівняно довгим машинним словом (32 розряди і більше).

Відносна адресація, або базування. Виконавська адреса визначається сумою адресного коду команди (АК) і деякого числа, яке має назву адреса бази (АБ):

$$ВА = АК + АБ.$$

Для зберігання базових адрес у машині можуть бути передбачені регістри або спеціально виділені для цієї мети елементи пам'яті (базові регістри). У команді виділяється поле *В* для вказівки номера базового регістра.

Відносна адресація дозволяє при меншій довжині адресного коду команди забезпечити доступ до будь-якого елементу пам'яті. Для цього число розрядів у базовій адресі вибирають таким, щоб можна було адресувати будь-яку комірку ОП, а адресний код (АК) самої команди використовують для уявлення лише порівняно короткого «зсуву» (позначають буквою *D*). Зсув *D* визначає положення операнда відносно початку масиву, що задається базовою адресою (АБ).

Підсумовування при утворенні ВА пов'язане з втратою часу. Тому застосовують також формування виконавської адреси методом поєднання. У цьому випадку базова адреса містить старші, а зсув – молодші розряди виконавської адреси, які об'єднуються в регістрі адреси пам'яті згідно з операцією конкатенації.

Проте при поєднанні базова адреса може задавати не будь-яку комірку, а тільки ті, адреси яких містять 0 у молодших розрядах.

Відносна адресація забезпечує так звану переміщуваність програм, тобто можливість пересування програм у пам'яті без змін усередині самої програми.

Укорочена адресація. Для зменшення довжини коду команди часто застосовується так звана укорочена адресація. Суть її зводиться до того, що в команді задаються тільки молодші розряди адрес, старші розряди при цьому мають на увазі нульовими. Така адресація дозволяє використовувати тільки

невелику групу фіксованих комірок із початковими (короткими) адресами, і тому може застосовуватися лише спільно з іншими способами адресації.

Регістрова адресація є окремим випадком укороченої, коли в якості фіксованих комірок із короткими адресами використовуються регістри процесора. Наприклад, якщо таких регістрів 16, то для адреси досить чотири двійкові розряди. Регістрова адресація разом зі скороченням довжини адрес операндів дозволяє збільшити швидкість виконання операцій, оскільки зменшується число звернень до ОП.

Непряма адресація. Адресний код команди вказує адресу елементу пам'яті, у якій знаходиться адреса операнда або команди. Таким чином, непряма адресація може бути інакше визначена як адресація адреси.

На непряму адресацію вказує код операції команди, а в деяких ЕОМ у команді відводиться спеціальний розряд (показчик адресації), і цифра 0 або 1 в ньому вказує, є адресна частина команди прямою адресою або непрямою.

У деяких ЕОМ використовується багатоступінчата непряма адресація. У цьому випадку елементи пам'яті містять також розряд – показчик непрямої адресації. Якщо цей розряд вказує на продовження непрямої адресації, то машина послідовно вибирає з пам'яті адреси до тих пір, поки не буде знайдена комірка, у якій розряд-показчик визначить пряму адресацію. Адреса з цієї останньої комірки і є виконавською адресою.

Непряма адресація широко використовується в мікроконтролерах, що мають коротке машинне слово, для подолання обмежень короткого формату команди.

Автоінкрементна і автодекрементна адресації. Оскільки регістрова непряма адресація вимагає попереднього завантаження регістра з ОП непрямою адресою, що пов'язане з втратою часу, такий тип адресації особливо ефективний при обробленні масиву даних, якщо є механізм автоматичного приросту або зменшення вмісту регістра при кожному зверненні до нього. Такий механізм має назву відповідно автоінкрементна і автодекрементна адресація. У цьому випадку достатньо один раз завантажити в регістр адресу першого оброблюваного елементу масиву, а потім при кожному зверненні до регістра в ньому в результаті інкрементної (декрементної) процедури формується адреса наступного елементу масиву.

При автоінкрементній адресації спочатку вміст регістра використовується як адреса операнда, а потім отримує приріст, рівний числу байт в елементі масиву. При автодекрементній адресації спочатку вміст вказаного в команді регістра

зменшується на число, рівне числу байт в елементі масиву, а потім використовується як адреса операнда.

Автоінкрементна і автодекрементна адресації можуть розглядатися як спрощений варіант індексації – дуже важливого механізму перетворення адресних частин команд і організації обчислювальних циклів, тому їх часто називають автоіндексацією.

Адресація слів змінної довжини. Ефективність обчислювальних систем, призначених для оброблення даних (економічних, планових та ін.), підвищується, якщо є можливість виконувати операції зі словами змінної довжини. У цьому випадку в машині повинна бути передбачена адресація слів змінної довжини, яка зазвичай реалізується шляхом вказівки в команді місцеположення в пам'яті початку слова і його довжини.

Зазвичай в ЕОМ одночасно використовується декілька типів адресації. Тип адресації вказується або неявно кодом операції, або в явній формі в спеціальному полі адресної частини команди.

3.11. АДРЕСАЦІЯ СТЕКА

Стекова пам'ять, що реалізовує безадресне задання операндів, є ефективним елементом сучасної архітектури ЕОМ. Враховуючи своєрідність стекової адресації, її розгляд виділений в окремий підрозділ.

Нагадаємо, що стек є групою послідовно пронумерованих регістрів (апаратний стек) або елементів пам'яті, забезпечених показчиком стека (зазвичай регістром), у якому автоматично при записі і читанні встановлюється номер (адреса) останньої зайнятої комірки стека (вершини стека). При операції запису слово даних, що заноситься в стек, поміщається в наступну за порядком вільну комірку стека, а при прочитуванні зі стека витягується останнє слово, що надійшло в нього. Таким чином, у стеку реалізується правило «останній прийшов – перший вийшов».

Указане правило при зверненні до стека реалізується автоматично, і тому при операціях зі стеком можливе безадресне задання операнда – команда не містить адреси комірки стека, але містить адресу (або вона мається на увазі) елементу пам'яті або регістра, звідки слово передається в стек або куди поміщається зі стека.

При виконанні команди передання в стек слова з регістра або комірки ОП спочатку показчик стека (ПСт) збільшується на 1, а потім слово поміщається в комірку стека з номером, що зберігається в ПСт. При команді читання зі стеку слова даних (перенос у регістр або комірку ОП) спочатку слово витягується

з вершини стека, а потім покажчик стека ПСт зменшується на 1. Як це не здається на перший погляд дивовижним, але при відповідному розташуванні операндів у стеку можна програмувати обчислювальні алгоритми повністю безадресними командами, вказуючими тільки тип операції. Така команда відповідно коду операції витягує зі стека один або два операнди, виконує над ними необхідну операцію (на рис. 3.4. наведено приклад операції складання) і заносить результат у стек.

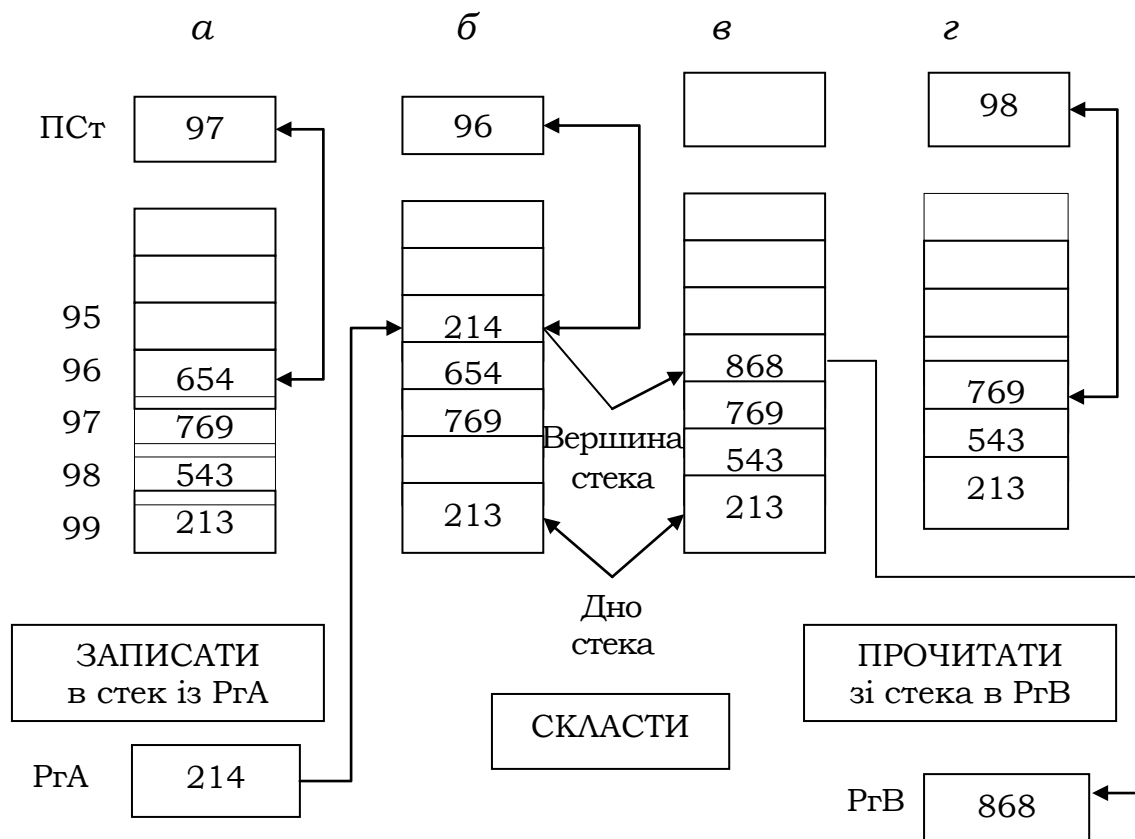


Рис. 3.4. Стекова адресація:

а – початковий стан стека, покажчик стека ПСт = 97; б – стек після виконання команди «ЗАПИСАТИ в стек із RгA»; в – стек після виконання команди «СКЛАСТИ»; г – стек після виконання команди «ПРОЧИТАТИ зі стека в RгB»

Безадресні команди на основі стекової адресації гранично скорочують формат команд, економлять пам'ять і сприяють підвищенню продуктивності ЕОМ. Проте при використанні безадресних команд виникають ускладнення з побудовою команд передання управління і команд для роботи з периферійними пристроями.

У сучасній архітектурі мікропроцесорів стек і стекова адресація широко використовуються при організації переходів до підпрограм і поверненні з них, а також у системах переривань.

3.12. КОМАНДИ, ПРОЦЕДУРИ І МІКРОПРОГРАМИ ПЕРЕДАННЯ УПРАВЛІННЯ

Розглянемо команди, які керують порядком виконання команд програми.

У звичайному випадку після виконання чергової команди вибирається команда, розташована в наступному за порядком елементі пам'яті. При цьому адреса команди зберігається в спеціальному реєстрі, який має назву лічильник команд (ЛК), вміст якого після виконання кожної команди автоматично збільшується на 1, а якщо пам'ять має побайтну адресацію, то вміст збільшується на стільки, скільки байт містить поточна команда (*приріст адреси команди ЛК*).

Звичайний порядок виконання команд може бути порушений:

1) командами переходу (командами передання управління);
2) командами заміщення (виконання); 3) зміною стану програми; 4) запитами програми на переривання.

Команди переходу (передання управління). Відомі численні варіанти команд переходу, проте загальний принцип полягає в тому, що адресна частина команди переходу безпосередньо або після підсумовування з вмістом базового реєстра передається в лічильник команд, і, таким чином, після даної команди може бути виконана команда з довільного елементу пам'яті, номер якої визначається адресною частиною команди переходу.

При *умовному переході* адреса наступної команди залежить від виконання деякої умови. Зазвичай якщо умова виконується, відбувається перехід до команди за адресою, вказаною в адресній частині команди умовного переходу. Інакше вибирається наступна за порядком команда, адреса якої визначається, як завжди, вмістом ЛК, збільшеним на приріст адреси команди ЛК. Так само як і при безумовному переході, у команді умовного переходу можуть використовуватись відносна і непряма адресації.

Звичайне виконання машинної команди супроводжується виробленням кодів ознаки результату, що зберігаються в спеціальному реєстрі ознак, або прапорів. Команда умовного переходу аналізує сформовану попередньою командою ознаку. Так, команди умовних переходів дозволяють реалізувати програми з розгалуженнями залежно від проміжних результатів обчислень або стану машини.

Важливим випадком передання управління є *безумовні переходи до підпрограм*. Їх особливість полягає в тому, що, крім переходу, вони повинні забезпечити після закінчення підпрограми повернення до початкової програми, до тієї її команди,

звідки був здійснений перехід. Зазвичай для переходів до підпрограм використовується спеціальна команда «Безумовний перехід із поверненням».

«Операція заміщення», яка реалізується командою «Виконання», замість чергової команди виконує команду «Заміщення», адреса якої зазначена в адресній частині команди «Виконання». Якщо команда «Заміщення» не виявилась командою переходу, відтворюється природний порядок виконання команд програми. У випадку операції «Заміщення» команда «Виконання» зберігає незмінним значення лічильника команд.

Питання для самоперевірки

1. Із яких основних елементів складається мікропроцесорний пристрій?
2. Визначте мікропроцесорну систему.
3. Що означає термін «прямий доступ до пам'яті»?
4. Які типи пам'яті використовуються в МПС?
5. Що таке контролер?
6. Яке призначення має системна шина?
7. Як функціонує мікропроцесорна система?
8. Які є способи передання інформації в МПС?
9. Що таке командний цикл?
10. Наведіть класифікацію АЛП.
11. Поясніть зв'язок між архітектурою МПС та складом її команд.
12. Як виглядає формат команди мікропроцесора?
13. Наведіть класифікацію команд мікропроцесора?
14. Охарактеризуйте способи адресації операндів у командах мікропроцесора.

РОЗДІЛ 4

МІКРОКОНТРОЛЕРИ

Мікроконтролери – це перша сходинка на шляху, що пройшли інформаційні технології до сучасних суперкомп'ютерів, але сьогодні можливості мікроконтролерів та їх роль у розвитку цифрової техніки важко переоцінити.

У цьому розділі наведена загальна інформація про мікроконтролери, їх структуру і організацію. Розділ включає короткий опис архітектури мікроконтролерів, зокрема мікроконтролерів фірми Intel сімейства 8051 (MCS-51) і мікроконтролерів фірми Atmel сімейства AVR.

4.1. ЦИФРОВІ ПРИСТРОЇ НА ОСНОВІ МІКРОКОНТРОЛЕРІВ

Еволюція мікропроцесорних систем рухалась у напрямку зменшення розмірів цифрових пристроїв, що спонукало розробляти нові технології щодо виготовлення електронних компонентів. У процесі мініатюризації функціональні блоки, які у звичайних комп'ютерних системах розміщувалися на окремих інтегральних схемах (процесор, пам'ять, порти введення/виведення, таймери, контролери переривань та ін.) стали інтегрувати в одному кристалі. Так з'явилися мікроконтролери, або, як їх прийнято називати у вітчизняній літературі, однокристалні ЕОМ (або міні ЕОМ). Такий підхід дозволив значно скоротити не лише розміри цифрових пристроїв, але й їх складність, терміни розроблення і ціну. Скорочення числа компонентів спричинило підвищення надійності готових пристроїв. У деяких випадках для побудови повноцінної вбудованої мікропроцесорної системи досить однієї мікросхеми – мікроконтролера.

Таким чином, архітектура мікроконтролера – це результат еволюції архітектури мікропроцесорів і мікропроцесорних систем у напрямку істотного зниження ціни цифрового пристрою та уникнення зайвих апаратних витрат. Мікроконтролер є мікропроцесорною системою, яка виготовлена як окрема інтегральна схема (див. рис. 4.1).

Ця інтегральна схема включає все, що необхідно для функціонування мікропроцесорної системи, тобто мінімальну конфігурацію: процесор, запам'ятовуючий пристрій, регістр команд, внутрішній генератор тактових сигналів. До складу мікроконтролера також входить електронна схема, що програмується, для зв'язку із зовнішнім середовищем. Обґрунтовано це тією обставиною, що мікроконтролери мають незначну за об'ємом пам'ять, яка розділяється на пам'ять програм і пам'ять даних, а також спрощену й орієнтовану на виконання нескладних задач управління системою команд і даних.

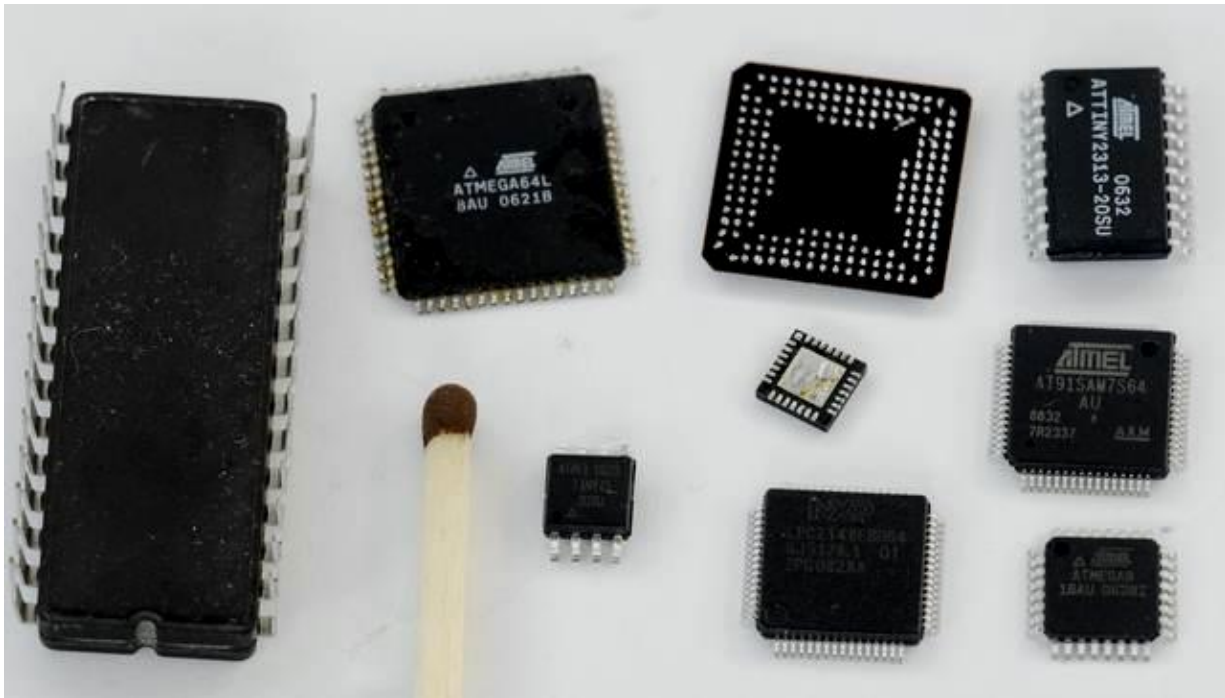


Рис. 4.1. Сучасні мікроконтролери в порівнянні з розміром сірника

Наприклад, мікроконтролери використовуються для прокручування касети в магнітофоні, вибирають доріжку на компакт-диску, підтримують температуру в холодильнику і т. ін.

До мікроконтролерів і однокристальних мікро ЕОМ ми відносимо сьогодні ті перші процесори, на основі яких створювалися перші персональні комп'ютери. Зменшившись із часом у розмірах і набувши функціонального різноманіття, ці процесори в новій якості слугують цілям оброблення інформації і вирішення задач управління.

Використання однокристальних мікроконтролерів дозволяє розробнику цифрового пристрою втілити знайдене рішення конкретної, специфічної задачі при мінімальних витратах щодо ціни і розміру.

Швидкий розвиток і поширення однокристальних мікроконтролерів обумовлені не тільки їх дешевизною, а й універсальністю. Сучасні мікроконтролери мають усі ресурси для вирішення задач управління, при цьому різноманітність їх зразків дозволяє задовольнити будь-які потреби. Зараз саме вони є прикладом реалізації модульного підходу до побудови цифрових пристроїв.

Переважає більшість мікроконтролерних систем використовуються як конструктивна складова автомобілів, телефонів, систем і блоків для комп'ютерних систем, вимірювальних і контролюючих пристроїв, виробничих приладів. Такі мікроконтролерні системи називаються вбудованими (англ. *embedded*). Незважаючи на дуже потужне оснащення деяких із них, більшість мають мінімальні габарити і працюють за нескладними

програмами. Зазвичай програмне забезпечення навіть не має операційної системи, тому що її застосування не виправдане для виконання простих завдань.

Перелік пристроїв, побудованих на базі мікроконтролерів або в складі яких є мікроконтролер, може бути нескінченним. Це мікро – АТС, автовідповідачі, пристрої для автоматичного визначення номеру, мобільні телефони, зарядні пристрої, факси, модеми, пейджери, таймери, системи сигналізації, вимірювальні прилади, лічильники води, газу й електроенергії, дозиметри, прилади автосигналізації, системи керування запалюванням і упорядкуванням палива, панелі управління пристроями, радарні детектори, інтелектуальні датчики, системи керування електро-двигунами, промислові роботи, регулятори температури, вологості, тиску, схеми управління аудіосистемами, системи синтезу мовних повідомлень, відеоігри, системи дистанційного керування, касові апарати і т. ін. Очевидно, мікроконтролери використовуються в ПК та комп'ютерних периферійних пристроях.

Перспективи щодо різноманіття цифрових пристроїв, побудованих на базі мікроконтролерів, важко переоцінити. Нові цифрові пристрої та модифікації існуючих з'являються майже щоденно, галузі їх використання розширюються.

4.2. КЛАСИФІКАЦІЯ І СТРУКТУРА МІКРОКОНТРОЛЕРІВ

Сьогодні виготовляється кілька типів мікроконтролерів. Наведемо їх класифікацію, але зазначимо, що вона досить умовна.

Мікроконтролери розділяють на три основні класи:

- 1) 8-розрядні мікроконтролери для вбудованих систем;
- 2) 16- і 32-розрядні мікроконтролери;
- 3) цифрові сигнальні процесори (DSP).

Найбільш поширеним представником сімейства мікроконтролерів є 8-розрядні пристрої, які використовуються в промисловості, побутовій і комп'ютерній техніці. Вони пройшли у своєму розвитку шлях від простих пристроїв із відносно слаборозвиненою периферією до сучасних багатофункціональних контролерів, що забезпечують реалізацію складних алгоритмів управління в реальному масштабі часу. Причиною життєздатності 8-розрядних мікроконтролерів є використання їх для управління реальними об'єктами, де застосовуються в основному алгоритми з розгалуженнями, швидкість оброблення яких практично не залежить від розрядності процесора. Зростанню їх популярності сприяє також постійне розширення номенклатури цифрових пристроїв, які виготовляються такими відомими фірмами, як Motorola, Microchip, Intel, Zilog, Atmel і багатьма іншими.

Сучасні 8-розрядні мікроконтролери мають, як правило, ряд відмітних ознак.

Перерахуємо основні властивості мікроконтролерів:

1) модульна організація, при якій на базі одного процесорного ядра (центрального процесора) проектується ряд (лінійка) мікроконтролерів, які відрізняються об'ємом і типом пам'яті програм, об'ємом пам'яті даних, набором периферійних модулів, частотою синхронізації;

2) використання закритої архітектури, яка характеризується відсутністю виводів шин адреси і даних на корпусі мікроконтролера. Таким чином, мікроконтролер є закінченою системою оброблення даних, нарощування можливостей якої з використанням паралельних магістралей адреси і даних не передбачається;

3) можливості використання типових функціональних периферійних модулів (таймерів, датчиків, контролерів послідовних інтерфейсів, аналогово-цифрових перетворювачів та ін.);

4) можливості розширення режимів роботи мікроконтролерів за рахунок ініціалізації спеціальних функцій (програмування мікроконтролерів).

Розглянемо принцип побудови мікроконтролерів. Мікроконтролер складається з двох основних модулів: процесорного ядра і змінного функціонального блока. Усі мікроконтролери одного сімейства мають однакове процесорне ядро і змінний функціональний блок, який відрізняє мікроконтролери різних моделей.

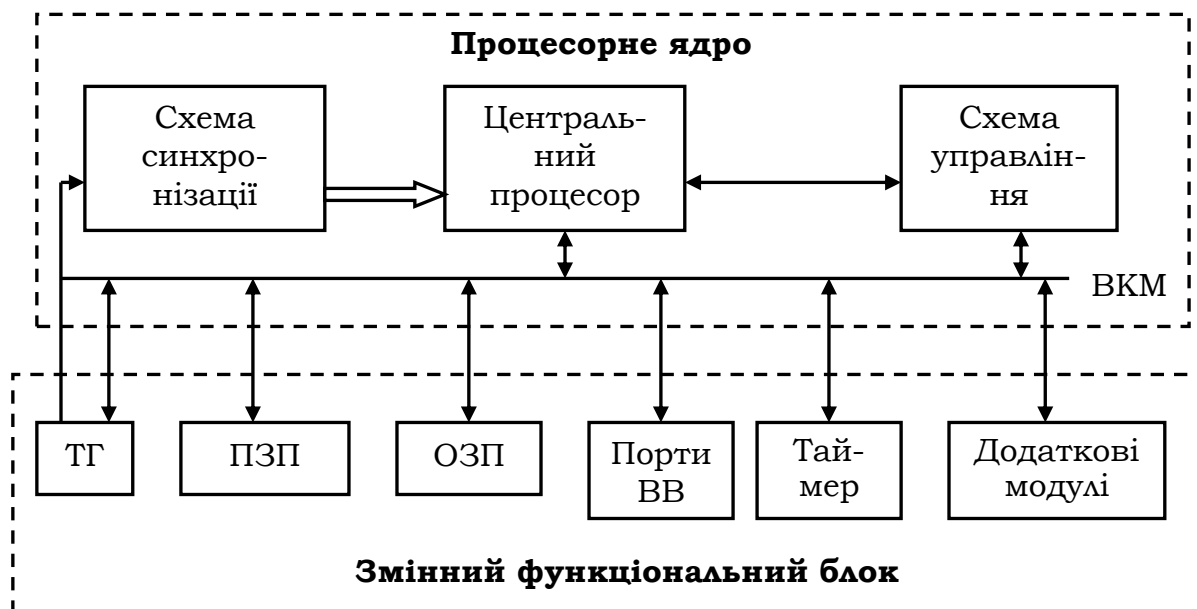


Рис. 4.2. Модульна організація мікроконтролера:

ТГ – тактовий генератор; ПЗП – постійний запам'ятовуючий пристрій;
ОЗП – оперативний запам'ятовуючий пристрій; порти ВВ – порти введення/виведення, ВКМ – внутрішня контролерна магістраль

Процесорне ядро включає:

- центральний процесор;
- внутрішню контролерну магістраль (ВКМ) у складі шин адреси, даних і управління;
- схему синхронізації;
- схему управління режимами роботи, яка підтримує режими зниженого енергоспоживання, початкового запуску (скидання) і т. ін. (див. рис 4.2).

Змінний функціональний блок включає модулі пам'яті різного типу і об'єму, порти введення/виведення, модулі тактових генераторів (ТГ), таймери. У відносно простих мікроконтролерах модуль оброблення переривань входить до складу процесорного ядра. У складніших МК він є окремим модулем із розвиненими можливостями. До складу змінного функціонального блока можуть входити і такі додаткові модулі, як компаратори напруги, аналого-цифрові перетворювачі (АЦП) та ін. Кожен модуль проектується для роботи у складі МК з урахуванням протоколу ВКМ. Цей підхід дозволяє створювати різноманітні за структурою мікроконтролери в межах одного сімейства.

4.3. ПРОЦЕСОРНЕ ЯДРО МІКРОКОНТРОЛЕРА ТА ЙОГО СТРУКТУРА

Основними характеристиками, які визначають продуктивність процесорного ядра мікроконтролера, є:

- набір регістрів для зберігання проміжних даних;
- система команд процесора;
- способи адресації операндів у просторі пам'яті;
- організація процесів вибірки і виконання команди.

З точки зору системи команд і способів адресації операндів процесорне ядро сучасних 8-розрядних МК реалізує один із двох принципів побудови процесорів:

- процесори з CISC-архітектурою, що реалізують так звану повну систему команд (Complicated Instruction Set Computer);
- процесори з RISC-архітектурою, що реалізують скорочену систему команд (Reduced Instruction Set Computer).

CISC-процесори мають великий набір команд із розвиненими можливостями адресації, що дає розробникові змогу вибрати необхідний набір команд для виконання певної операції. Процесор, який використовується у 8-розрядному мікроконтролері з CISC-архітектурою, може мати одnobайтовий, двобайтовий і трибайтовий (рідко чотирибайтовий) формат команд. Час виконання команди може складати від 1 до 12 циклів. До мікроконтролерів з CISC-архітектурою відносяться:

- мікроконтролери фірми Intel з ядром MCS-51, які підтримуються нині цілим рядом виробників периферійних модулів;

– мікроконтролери сімейств HC05, HC08 і HC11 фірми Motorola і ряд інших.

У процесорах із RISC-архітектурою набір команд скорочений до мінімуму. Для реалізації складніших операцій доводиться комбінувати команди. При цьому всі команди мають формат фіксованої довжини (наприклад, 12, 14 або 16 біт). Вибірка команди з пам'яті і її виконання здійснюється за один цикл (такт) синхронізації. Система команд RISC-процесора припускає можливість рівноправного використання всіх регістрів процесора. Це забезпечує додаткову гнучкість при виконанні ряду операцій.

До мікроконтролерів з RISC-процесором відносяться мікроконтролери AVR фірми Atmel, мікроконтролери PIC16 і PIC17 фірми Microchip та ін.

На перший погляд, мікроконтролери з RISC-процесором повинні мати вищу продуктивність у порівнянні з CISC-мікроконтролерами при одній і тій же тактовій частоті внутрішньої магістралі. Проте на практиці питання про продуктивність вирішується не так однозначно і є більш складним.

З точки зору організації процесів вибірки і виконання команди в сучасних 8-розрядних мікроконтролерах застосовується одна з класичних архітектур МПС. Одна з них має назву архітектура фон Неймана (або прінстонська). Іншу називають архітектурою Гарвардської лабораторії або просто гарвардською.

Основною особливістю фон-нейманівської архітектури є використання загальної пам'яті для зберігання програм і даних (див. рис. 4.3).

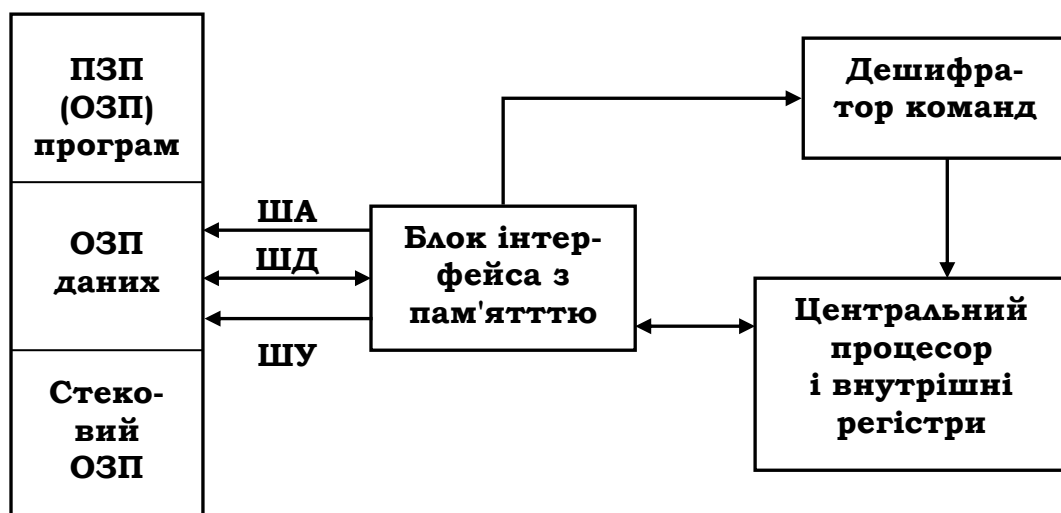


Рис. 4.3. Структура МПС з фон-нейманівською архітектурою:
ПЗП – постійний запам'ятовуючий пристрій; ОЗП – оперативний запам'ятовуючий пристрій; ША – шина адреси; ШД – шина даних; ШУ – шина управління

Основною перевагою мікроконтролерів з архітектурою фон Неймана – є спрощення конфігурації МПС, оскільки реалізується звернення тільки до однієї загальної пам'яті. Крім того, використання єдиної області пам'яті дозволило оперативно перерозподілити ресурси між областями програм і даних, що істотно підвищило гнучкість МПС з точки зору розробника програмного забезпечення. Розміщення стека в загальній пам'яті полегшує доступ до його вмісту. Тому не випадково фон-нейманівська архітектура стала основною архітектурою універсальних комп'ютерів, включаючи ПК.

Основною особливістю гарвардської архітектури є використання окремих адресних просторів для зберігання команд і даних (див. рис. 4.4).

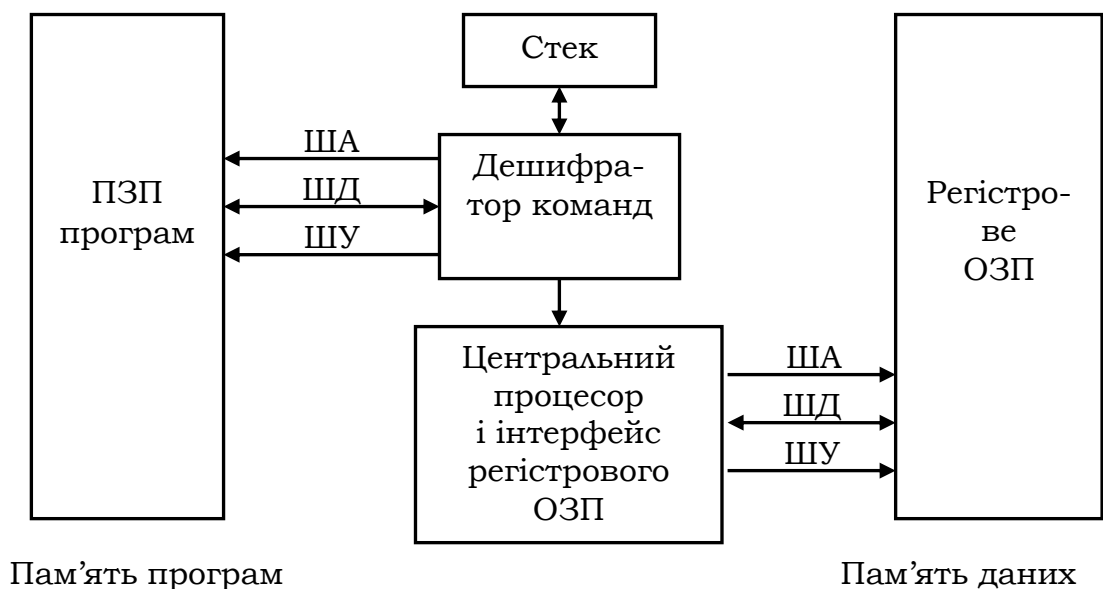


Рис. 4.4. Структура МПС із гарвардською архітектурою:
ПЗП – постійний запам'ятовуючий пристрій; ОЗП – оперативний запам'ятовуючий пристрій; ША – шина адреси; ШД – шина даних; ШУ – шина управління

Гарвардська архітектура майже не використовувалася до кінця 70-х рр. XX ст., доки розробники мікроконтролерів не зрозуміли, що вона дає певні переваги при проектуванні автономних систем управління.

Аналіз реальних програм роботи МПС показав, що необхідний об'єм пам'яті даних мікроконтролера, що використовується для зберігання проміжних результатів, як правило, на порядок менше необхідного об'єму пам'яті програм. Тому використання єдиного адресного простору призводило до збільшення формату команд за рахунок збільшення числа розрядів для адресації операндів. Застосування окремої невеликої за об'ємом пам'яті даних сприяло скороченню довжини команд і прискоренню

пошуку інформації в пам'яті даних. Тому досвід використання МПС для управління різними об'єктами та для реалізації більшості алгоритмів управління показав, що такі переваги фон-нейманівської архітектури, як гнучкість і універсальність, не мають великого значення. Крім того, гарвардська архітектура забезпечує потенційно вищу швидкість виконання програми в порівнянні з фон-нейманівською за рахунок можливості реалізації паралельних операцій. Вибірка наступної команди може відбуватися одночасно з виконанням попередньої, тому немає необхідності зупиняти процесор на час вибірки команди. Можливість виконання операцій у паралельному режимі дозволяє забезпечувати виконання різних команд за однакове число тактів, що спрощує визначення часу виконання циклів і критичних ділянок програми.

Більшість виробників сучасних 8-розрядних мікроконтролерів використовують гарвардську архітектуру. Проте гарвардська архітектура є недостатньо гнучкою для реалізації певних програмних процедур. Тому порівняння мікроконтролерів, виконаних за різними архітектурами, слід проводити з точки зору конкретного застосування.

4.4. ПРИКЛАДИ МІКРОКОНТРОЛЕРІВ

Нині найбільш яскравими представниками мікроконтролерів з CISC- та RISC-процесорами, що мають відповідно фон-нейманівську і гарвардську архітектуру, є мікроконтролери сімейства i8051 фірми Intel і AVR – мікроконтролери фірми Atmel, які за рядом характеристик перевершили дуже відомі RISC-мікроконтролери фірми Microchip. Розглянемо архітектуру і особливості вищеперелічених мікроконтролерів.

4.4.1. Мікроконтролер i8051

Однокристалний мікроконтролер i8051 (див. рис 4.5) уперше був виготовлений фірмою Intel у 1980 р. Цей мікроконтролер використовувався у вбудованих системах управління технологічними процесами і мав гарвардський тип архітектури. Пік популярності цього пристрою припадає на 90-ті р. XX ст. У теперішній час він вважається застарілим, і його замінили більш сучасні пристрої, у яких, однак, використовується ядро мікроконтролера 8051. Мікроконтролери з 8051-сумісними ядрами виготовляються більш ніж 20-ма незалежними фірмами, такими як Atmel, Maxim IC (дочірня компанія Dallas Semiconductor), NXP (раніше відома як Philips Semiconductor), Winbond, Silicon Laboratories, Texas Instruments і Cypress Semiconductor. Існує радянський аналог даного пристрою – мікроконтролер KP1816BE51. Офіційна назва 8051-сімейства мікроконтролерів Intel – MCS-51.



Рис. 4.5. Мікроконтролер i8051

Перші 8051-сімейства Intel виготовлялись із застосуванням n-МОН (рос. n-МОП, англ. n-MOS) технології, наступні версії, що мали букву «С» в назві, такі як 80C51, використовували КМОН-технологію та були більш енергозбережливі, ніж n-МОН попередники (це забезпечувало їм переваги при використанні в пристроях, які мали автономний елемент живлення).

Назва вищезгаданих технологій походить від назви МОН-транзистора (Метал – Оксид – Напівпровідник). Потреба енергоживлення у МОН-транзисторів дуже мала (особливо у р-МОН), що дозволяє на їх основі виготовляти пристрої, які потребують енергоекономії (часи, калькулятори та ін.). n-МОН схеми використовуються завдяки простоті виготовлення і трохи більшої швидкодії. Переваги МОН-технологій – це висока економічність, висока щільність розміщення на кристалі та низька ціна. КМОН (або К-МОН – комплементарна логіка на транзисторах Метал – Оксид – Напівпровідник; англ. CMOS – Complementary-symmetry/metal-oxide semiconductor) – це технологія побудови електронних схем. За технологією КМОН виготовляються мікросхеми, які практично не вживають електроенергію в статичному режимі. У більшості випадків можна вважати, що енергія вживається тільки під час перемикання станів. КМОН-схеми мають більш високу швидкодію і менше енергоживлення, але виготовляються за більш складним технологічним процесом та мають меншу щільність розміщення на кристалі. Більшість сучасних логічних мікросхем, у тому числі й мікропроцесорів, виготовляється за КМОН-технологією.

Мікроконтролери сімейства i8051 мають наступні апаратні особливості:

- внутрішній ОЗП об'ємом 128 байт;
- чотири двонаправлені побітні 8-розрядні порти введення/виведення, що налаштовуються (або програмуються);
- два 16-розрядні таймер-лічильники;
- вбудований тактовий генератор;
- адресний простір може сягати 64 Кб пам'яті програм і 64 Кб пам'яті даних;
- має дві лінії запитів на переривання від зовнішніх пристроїв;

– має послідовний інтерфейс для обміну інформацією з іншими мікроконтролерами або персональними комп'ютерами.

Мікроконтролер i8751 забезпечений ПЗП типу FLESH об'ємом 4 Кб.

Структурна схема мікроконтролера сімейства i8051

Мікроконтролер виготовляється із застосуванням високорівневої n-МОП технології. Через чотири програмовані паралельні порти введення/виведення і один послідовний порт мікроконтролер обмінюється даними із зовнішніми пристроями.

Сімейство MCS-51 має п'ять модифікацій із аналогічними характеристиками.

Система команд складається з 111 базових команд довжиною 1, 2 або 3 байти. У розпорядження програміста надається 32 регістра загального призначення (РЗП), 128 програмно-керованих прапорів та набір регістрів спеціальних функцій.

Основу структурної схеми (див. рис. 4.6) утворює внутрішня двонаправлена 8-бітова шина, яка зв'язує між собою основні вузли і блоки мікроконтролера: резидентну пам'ять програм, резидентну пам'ять даних, арифметико-логічний пристрій, регістри спеціальних функцій, пристрій управління і порти введення/виведення.

АСС – регістр акумулятора. У командах роботи з акумулятором використовується мнемоніка А. Наприклад, MOV A, P2. Мнемоніка АСС використовується для побітної адресації акумулятора. Наприклад, символічне ім'я п'ятого біта акумулятора в асемблері буде АСС.5.

Регістр В можна використовувати для операцій множення та ділення. В інших випадках його можна використовувати як додатковий регістр.

Регістр стану PSW використовується для зберігання інформації про стан поточної програми.

Регістр стека SP – це 8-бітний регістр, вміст якого інкрементується (набуває приросту або зменшення) перед зверненням до стека командами PUSH і CALL.

Показчик даних DPTR складається зі старшого DPH і молодшого DPL байтів. Він зберігає 16-бітну адресу зовнішньої пам'яті. Може використовуватись як 16-бітний регістр або як два 8-бітних.

Порт 0 ... Порт 3 обслуговуються регістрами спеціальних функцій P0, P1, P2, P3 відповідно, які є регістрами – «заскочками» портів P0, P1, P2, P3.

Буфер послідовного порту SBUF представляє собою два окремих регістра: буфер передавача і буфер приймача. Коли дані записуються в SBUF, вони надходять у буфер передавача

(запис байта SBUF автоматично ініціює його передання через послідовний порт). Коли дані прочитуються із SBUF, вони автоматично вибираються із буфера приймача.

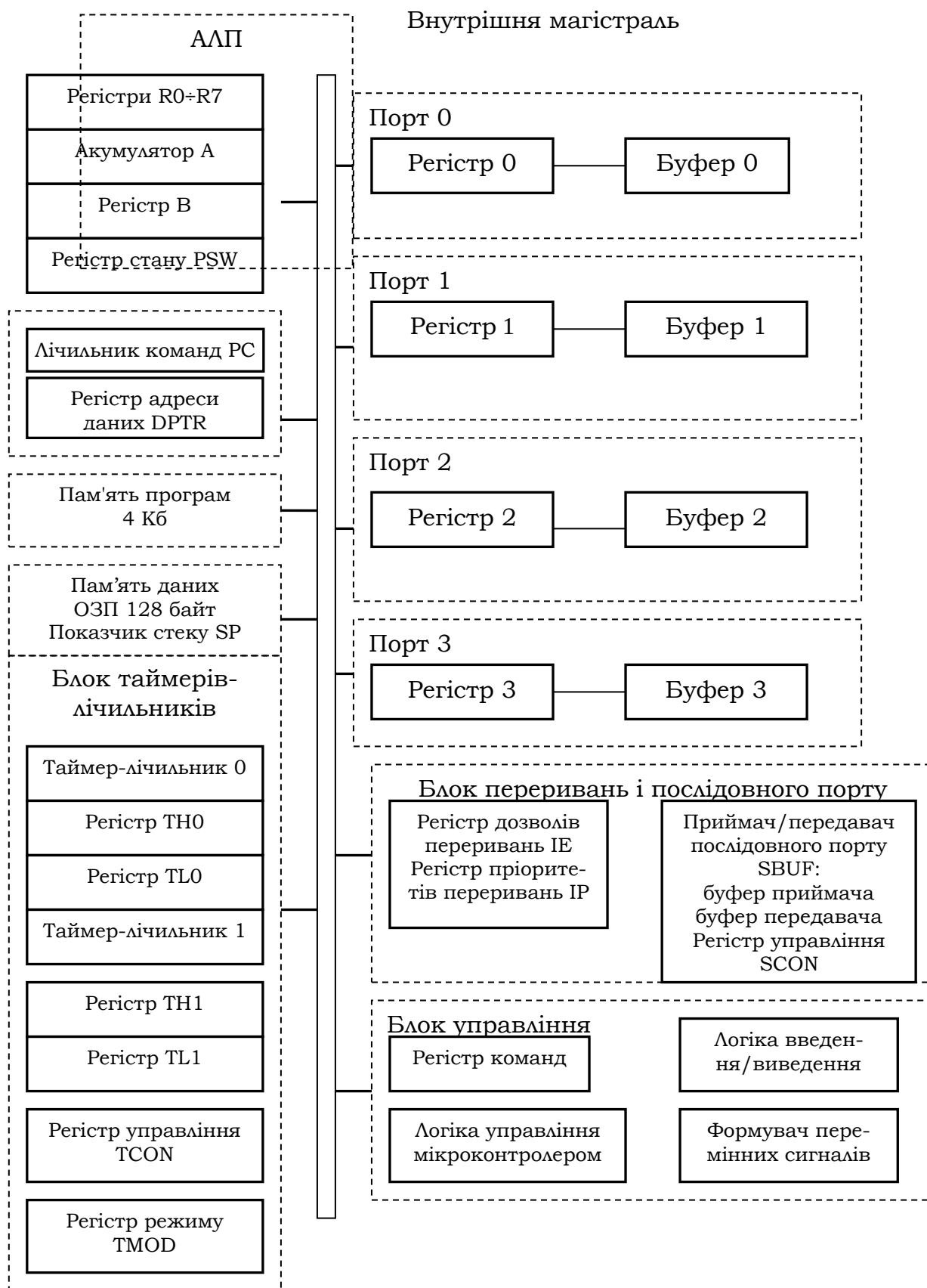


Рис. 4.6. Структурна схема i8051

Регістрові пари (TH0, TL0) і (TH1, TL1) створюють 16-бітні регістри таймера/лічильника 0 і таймера/лічильника 1.

До регістрів управління належать регістри спеціальних функцій IP, IE, TMOD, TCON, SCON і PCON, які зберігають біти управління та біти стану системи переривань, таймерів/лічильників та послідовного порту.

Наприклад, регістр управління споживанням PCON дозволяє зупиняти роботу мікроконтролера для зменшення споживання електроенергії та рівня перешкод від мікроконтролера. Ще більшого зменшення енергоспоживання та перешкод можна досягти за рахунок зупинки генератора мікроконтролера. Це досягається переключенням біта регістра управління споживанням PCON.

У таблиці 4.1 наведені адреси регістрів спеціальних функцій.

Таблиця 4.1

Регістри спеціальних функцій

Найменування	Призначення	Адреса
ACC	Акумулятор	0E0H
B	Регістр В	0F0H
PSW	Регістр стану програми	0D0H
SP	Показчик стека	81H
DPTR	Показчик адреси даних, 2 байти	
DPL	Молодший байт	82H
DPH	Старший байт	83H
P0	Порт 0	80H
P1	Порт 1	90H
P2	Порт 2	0A0H
P3	Порт 3	0B0H
IP	Регістр пріоритетів переривань	0B8H
IE	Регістр дозволів на переривання	0A8H
TMOD	Регістр режимів	89H
TCON	Регістр управління	88H
TH0	Таймер/лічильник 0. Старший байт	8CH
TL0	Таймер/лічильник 0. Молодший байт	8AH
TH1	Таймер/лічильник 1. Старший байт	8DH
TL1	Таймер/лічильник 1. Молодший байт	8BH
SCON	Регістр управління послідовним портом	98H
SBUF	Буфер послідовного порту	99H
PCON	Регістр управління споживанням	87H

Регістри ACC, B, PSW, P0, P1, P2, P3, IP, IE, TCON, SCON можуть адресуватись побітно.

Розглянемо особливості роботи АЛП мікроконтролера i8051. АЛП виконує стандартний набір арифметичних і логічних операцій над 8-бітовими даними. АЛП може виконувати арифметичні

операції «Складання», «Віднімання», «Множення» і «Ділення»; логічні операції «і», «або», «виключне або», а також операції циклічного зсуву, скидання, інвертування. До входів АЛП підключені програмно недосяжні регістри T1 і T2, призначені для тимчасового зберігання операндів, схема десяткової корекції (DCU) і схема формування ознак результату операції (PSW).

Для інкрементування (нaroщування) вмісту регістрів використовується звичайна операція «Складання». За допомогою цієї операції АЛП також здійснює зсув (RAR) регістра – показника даних і автоматичне обчислення наступної адреси в резидентній пам'яті програм. Проста операція «Віднімання» використовується в АЛП для декрементування регістрів і порівняння змінних.

Прості операції автоматично утворюють «тандеми» для виконання таких операцій, як інкрементування 16-бітових регістрових пар. В АЛП реалізується механізм каскадного виконання простих операцій для реалізації складних команд. Так, наприклад, при виконанні однієї з команд умовного передання управління за результатами порівняння в АЛП тричі інкрементується лічильник команд (PC), двічі проводиться читання з RDM, виконується арифметичне порівняння двох змінних, формується 16-бітова адреса переходу і приймається рішення про те, робити або не робити перехід за програмою. Усі перераховані операції виконуються усього лише за 2 мкс.

Важливою особливістю АЛП є його здатність оперувати не лише байтами, але й бітами. Окремі програмно доступні біти можуть бути записані, скинуті (до нульового стану), інвертовані, передані, перевірені і використані в логічних операціях. Ця здатність досить важлива, оскільки для управління об'єктами часто застосовуються алгоритми, що містять операції над вхідними і вихідними булевими змінними, реалізація яких засобами звичайних мікропроцесорів пов'язана з певними труднощами.

Таким чином, АЛП може оперувати чотирма типами інформаційних об'єктів: булевими (1 біт), цифровими (4 біта), байтовими (8 біт) і адресними (16 біт). АЛП вміє виконувати 51 різну операцію пересилання або перетворення вищезгаданих даних. Оскільки використовується 11 режимів адресації (7 для даних і 4 для адрес), то шляхом комбінування операцій і режимів адресації базове число команд 111 розширюється до 255 з 256 можливих при однобайтовому коді операцій.

Методи адресації і система команд

Розглянемо 4 основні способи адресації.

Регістрова адресація. Операнд перебуває в одному з регістрів загального призначення R0 ... R7, номер якого визначається розрядами RS0, RS1 регістра PSW. Номер регістра визначається трьома молодшими бітами байта коду операції (КІП – перший

байт команди). Регістрова адресація дозволяє в одному байті вказати код операції й адресу операнда. Оскільки операнд перебуває у внутрішній комірці пам'яті, то звертання до зовнішньої пам'яті даних не потрібне. Наприклад, формат команди додавання буде наступним: ADD A,Rn.

Пряма адресація. Операнд перебуває у внутрішній пам'яті даних, для адресації якої використовується окремий байт команди. Команди з використанням прямої адресації мають мінімум два байти, можуть адресувати тільки 256 комірки, звернення за операндом до зовнішньої пам'яті не потрібне. Формат команди додавання з прямою адресацією буде наступним: ADD A,direct.

Непряма регістрова адресація. Цей спосіб адресації використовує регістри R0 й R1 як покажчики. Регістри визначаються розрядами RS0, RS1 регістра PSW. Вміст цих регістрів використовується як адреса для звертання до комірки внутрішньої пам'яті даних, де зберігається операнд. Команда виходить однобайтною, номер регістра вказує молодший біт. Довжина команди при цьому відповідає довжині команди з використанням регістрової адресації, але непряма адресація дає доступ до всього простору внутрішньої пам'яті даних. Формат команди додавання з непрямою регістровою адресацією буде наступним: ADD A,@Ri.

Безпосередня адресація. Операнд перебуває в окремому байті команди. Він представляє собою константу. Таким чином, команди з використанням безпосередньої адресації мають мінімум два байти, звернення за операндом до зовнішньої пам'яті не потрібне. Формат команди додавання з безпосередньою адресацією буде наступним: ADD A,#data.

Система команд сімейства MCS-51

Усього в системі команд сімейства MCS-51 можна виділити 5 груп команд: 1) арифметичні команди; 2) логічні команди; 3) команди зсуву; 4) команди пересилання даних; 5) команди передання управління.

Для відтворення команд використовуються спеціальні позначення (див. табл. 4.2).

Таблиця 4.2

Позначення в командах

Позначення	Значення
A	Акумулятор
Rn	Поточний регістр
direct	Безпосередня 8-бітова внутрішня адреса комірки даних
@Ri	8-бітова комірка внутрішньої пам'яті даних, що непрямо адресується через регістр Ri (непряма регістрова адресація)
#data	8-бітове або 16-бітове безпосереднє дане

Для більш наочного подання основні команди наводяться з усіма можливими варіантами їхнього запису.

Арифметичні команди:

1) ADD A, <байт-джерело> – додавання, складає вміст акумулятора A з вмістом байта-джерела, залишаючи результат в акумуляторі:

ADD A,Rn;
ADD A,@Ri;
ADD A,(direct);
ADD A,<#data>;

2) ADDC A, <байт-джерело> – додавання з урахуванням переносу:

ADDC A,Rn;
ADDC A,@Ri;
ADDC A,(direct);

3) SUBB A, <байт-джерело> – віднімання з урахуванням прапора:

SUBB A,Rn;
SUBB A,@Ri;
SUBB A,(direct);

4) DA A – десяткова корекція при додаванні;

5) DEC <байт> – декремент, робить віднімання «1» із зазначеного операнда:

DEC A;
DEC Rn;
DEC @Ri;
DEC (direct);

6) INC <байт> – інкремент – виконує додавання «1» до зазначеної змінної:

INC A;
INC Rn;
INC @Ri;
INC (direct);

7) INC DPTR – команда «інкремент покажчика даних», виконує інкремент вмісту 16-бітового покажчика даних (DPTR);

8) DIV A,B – ділення;

9) MUL AB – множення.

Логічні команди:

1) ANL байт призначення, <байт джерела> – «логічне «і» для змінних» виконує операцію логічного «і» над бітами зазначених змінних і поміщає результат в байти призначення:

ANL A,Rn;
ANL A,@Ri;
ANL A,;

ANL A,<# data>;

ANL ,A;

ANL ,#data;

2) ORL байт призначення, <байт джерела> – «логічне «або» для байтових змінних» виконує операцію логічного «АБО» над бітами зазначених змінних, записуючи результат у байт призначення:

ORL A ,Rn;

ORL A ,@Ri;

ORL A ,(direct);

ORL A ,<#data>;

ORL ,A;

3) ORL C, <біт джерела> – команда «логічне « » для бітових змінних» встановлює прапор переносу C в «1», якщо булева величина дорівнює логічній «1», у протилежному випадку встановлює прапор C в «0»:

ORL C,bit;

4) XRL <байт призначення>, <байт джерела> – команда «логічне «виключне або» для байтових змінних» виконує операцію «виключне або» над бітами зазначених змінних, записуючи результат у байт призначення:

XRL A, Rn;

XRL A,;

XRL A, @Ri;

XRL A, #data;

XRL ,A;

XRL ,#data;

5) CLR A – очищення акумулятора;

6) CPL A – інверсія акумулятора, кожен біт акумулятора інвертується.

Команди зсуву:

1) RL A – команда зсуву вмісту акумулятора вліво;

2) RLC A – команда зсуву вмісту акумулятора вліво через прапор переносу зсуває вісім бітів акумулятора вліво і вставляє прапор переносу у вивільнений розряд;

3) RR A – команда зсуву вмісту акумулятора вправо зсуває вправо на один біт усі вісім бітів акумулятора.

4) RRC A – команда зсуву вмісту акумулятора вправо через прапор переносу зсуває вісім бітів акумулятора вправо і вставляє прапор переносу у вивільнений розряд.

Команди пересилання даних:

1) MOV <байта призначення>, <байта-джерела> – пересилає байт, зазначений у другому операнді, у комірку, зазначену в першому операнді:

MOV A,Rn;

```
MOV A,(DIRECT);  
MOV A,#data;  
MOV A,Ri;  
MOV Rn,A;  
MOV Rn,(DIRECT);  
MOV Rn,#data;
```

2) MOV <біт призначення>, <біт джерела> – команда «переслати біт даних» в бітову змінну, зазначену в другому байті. Копіює біт у розряд, зазначений у першому операнді:

```
MOV C ,(BIT);  
MOV (BIT),C;
```

3) Команда MOV DPTR,#<data16> – завантажити покажчик даних 16-бітовою константою;

4) Команда MOVC A,@A+() – команда «переслати байт із пам'яті програм» – завантажує акумулятор байтом коду або константою з пам'яті програми:

```
MOVC A,@A+DPTR;  
MOVC A,@A+PC;
```

5) Команда MOVX <байт приймача>, <байт джерела> – команда «переслати в зовнішню пам'ять (із зовнішньої пам'яті) даних» – пересилає дані між акумулятором і байтом зовнішньої пам'яті даних:

```
MOVX A,@Ri;  
MOVX A,@DPTR;  
MOVX @ DPTR,A;  
MOVX @Ri,A;
```

6) SWAP A – команда «обмін тетрадами усередині акумулятора» – здійснює обмін між молодшими чотирма й старшими чотирма бітами акумулятора;

7) PUCH – команда «запис у стека» – збільшує покажчик стека на одиницю й після цього вміст зазначеної змінної копіюється в комірку внутрішньої пам'яті, що адресується за допомогою покажчика стека;

8) POP – команда «читання зі стека» – зчитує вміст комірки, що адресується за допомогою покажчика стека, у комірку пам'яті, при цьому покажчик стека зменшується на одиницю.

9) XCH A, <байт> – команда «обмін вмісту акумулятора зі змінним байтом» – здійснює обмін вмісту акумулятора із вмістом джерела, зазначеним у команді:

```
XCH A,(direct);  
XCH A,Rn;  
XCH A,@Ri;
```

10) XCHD A,@Ri – команда «обмін тетрадою» – виконує обмін молодшої тетради (біти 3 ... 0) акумулятора із вмістом молодшої тетради (біти 3 ... 0) комірки пам'яті.

Команди передавання управління:

1) CJNE <байт призначення>, <байт джерело>, <мітка> – команда порівняння й переходу, якщо не дорівнює:

CJNE A ,<direct>,<мітка>;

CJNE A ,#data,<мітка>;

CJNE Rn ,#data,<мітка>

CJNE @Ri ,#data,<мітка>;

2) AJMP<мітка> – абсолютний перехід;

3) DJNZ<байт>,<мітка> – декремент і перехід, якщо не дорівнює нулю:

DJNZ Rn,<мітка>;

DJNZ,<мітка>;

4) JB (bit) – перехід, якщо біт встановлений в «1»;

5) JNB (bit) – перехід, якщо біт не встановлений;

6) JBC(bit) – перехід, якщо біт встановлений, і скидання цього біта;

7) JC – перехід, якщо біт переносу встановлений;

8) JNC – перехід, якщо біт переносу не встановлений;

9) JMP @A+DPTR – непрямої перехід;

10) JNZ – перехід, якщо вміст акумулятора не дорівнює нулю;

11) JZ – перехід, якщо вміст акумулятора дорівнює нулю;

12) ACALL – абсолютний виклик підпрограми;

13) LCALL – команда «довгий виклик» – викликає підпрограму, що перебуває за зазначеною адресою;

14) LJMP – команда «довгий перехід» – виконує безумовний перехід за зазначеною адресою;

15) NOP – команда «немає операції» – виконує холостий хід і не впливає на регістри й прапори, крім як на лічильник команд (PC);

16) RET – команда «повернення з підпрограми» – послідовно вивантажує старший і молодший байти лічильника команд зі стека, зменшуючи покажчик стека на 2;

17) RETI – команда «повернення з переривання»;

18) SJMP <мітка> – команда «короткий перехід» – виконує безумовне розгалуження в програмі за зазначеною адресою.

4.4.2. Структурна організація AVR-мікроконтролера

Мікроконтролери фірми AVR функціонують у широкому діапазоні живлячої напруги: від 1,8 до 6,0 В. Енергоспоживання в активному режимі залежить від величини напруги живлення, від частоти, на якій працює AVR і від конкретного типу мікроконтролера. Детальні специфікації зазвичай наводяться в оригінальній технічній документації Atmel Corporation. Температурні діапазони роботи мікроконтролерів AVR для комерційних моделей містяться в діапазоні від 0° С до 70° С, для індустріальних – від -40° С до +85° С. На жаль, корпорація Atmel не випускає і не планує випускати AVR для роботи в

автомобільному ($-40^{\circ}\text{C} \dots +125^{\circ}\text{C}$) і військовому ($-55^{\circ}\text{C} \dots +125^{\circ}\text{C}$) температурних діапазонах.

З точки зору програміста, AVR є 8-розрядним RISC-мікроконтролером, що має швидкий гарвардський процесор, пам'ять програм, пам'ять даних, порти введення/виведення і різні інтерфейсні схеми. Гарвардська архітектура AVR реалізує повний логічний і фізичний розподіл не лише адресних просторів, але й інформаційних шин для звернення до пам'яті програм і до пам'яті даних, причому способи адресації і доступу до цих масивів пам'яті також різні. Подібна побудова вже ближче до структури цифрових сигнальних процесорів і забезпечує істотне підвищення продуктивності. Центральний процесор працює одночасно як із пам'яттю програм, так і з пам'яттю даних. Розрядність шини пам'яті програм розширена до 16 бітів.

Наступним кроком на шляху збільшення швидкодії AVR є використання технології конвеєризації, внаслідок чого цикл «вибірка – виконання» команди помітно скорочений. Наприклад, у мікроконтролерів сімейства i8051 коротка команда виконується за 12 тактів генератора (1 машинний цикл), впродовж якого процесор послідовно прочитує код операції і виконує її. У PIC-контролерах фірми Microchip, де реалізований конвеєр, коротка команда виконується впродовж 8 тактів (2 машинні цикли). За цей час послідовно дешифрується і прочитується код операції, виконується команда, фіксується результат і одночасно прочитується код наступної операції (однорівневий конвеєр). Тому в загальному потоці команд одна коротка команда реалізується за 4 періоди тактової частоти або за один машинний цикл.

У мікроконтролерах AVR теж використовується однорівневий конвеєр при зверненні до пам'яті програм і коротка команда в загальному потоці виконується, як і в PIC-контролерах, за один машинний цикл. Головна ж відмінність полягає в тому, що цей цикл у AVR складає всього один такт. Порівняймо часові діаграми виконання типової команди для різних мікроконтролерних платформ (див. рис. 4.7).

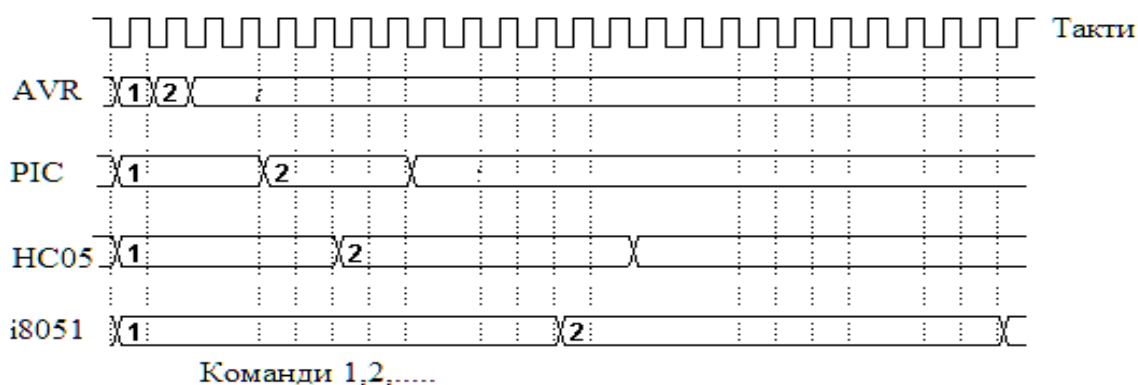


Рис. 4.7. Часові діаграми мікроконтролерних платформ

Відмінною рисою архітектури мікроконтролерів AVR є регістровий файл швидкого доступу, структура якого показана на рис. 4.8. Кожен із 32-х регістрів загального призначення завдовжки 1 байт безпосередньо зв'язаний з арифметико-логічним пристроєм (ALU) процесора. Іншими словами, в AVR існує 32 регістри – акумулятори. Ця обставина дозволяє в поєднанні з конвеєрним обробленням виконувати одну операцію в ALU за один машинний цикл. Так, два операнди витягаються з регістрового файлу, виконується команда, результат записується назад у регістровий файл впродовж тільки одного машинного циклу.

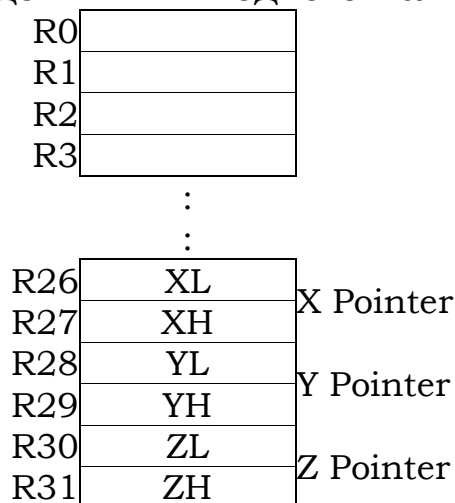


Рис. 4.8. Регістровий файл AVR

Шість із 32-х регістрів файлу можуть використовуватися як регістрові пари (X Pointer, Y Pointer, Z Pointer). Ці пари регістрів використовуються як 16-розрядні покажчики адреси при непрякій адресації даних. Один із цих покажчиків (Z Pointer) використовується також для доступу до даних, записаних у пам'яті програм мікроконтролера. Наявність трьох 16-бітових покажчиків адреси (X, Y і Z Pointers) істотно підвищує швидкість пересилання даних при роботі прикладної програми.

Регістровий файл займає молодші 32 байти в загальному адресному просторі SRAM мікроконтролера AVR. Таке архітектурне рішення дозволяє мати доступ до швидкої «регістрової» оперативної пам'яті мікроконтролера як безпосередньо в коді команди (безпосередня адресація), так і іншими способами адресації комірок SRAM. У технічній документації фірми Atmel ця корисна властивість носить назву «Швидке контекстне перемикання» і є ще однією відмінною особливістю архітектури AVR мікроконтролера, яка підвищує ефективність і продуктивність. Особливо помітна ця перевага при реалізації процедур цілочисельної 16-бітової арифметики, коли здійснюються багатократні пересилки між різними елементами пам'яті даних при оброблення арифметичних операндів в ALU.

Система команд AVR дуже розвинена і налічує до 133 різних інструкцій. Майже всі команди мають фіксовану довжину в одне слово (16 біт), що дозволяє в більшості випадків об'єднувати в одній команді і код операції, і операнд(и). Лише небагато команд мають розмір у 2 слова (32 біт) і відносяться до групи команд виклику процедури CALL, довгих переходів у межах усього адресного простору JMP, повернення з підпрограм RET і команд роботи з пам'яттю програм LPM. Розрізняють п'ять груп команд AVR: умовного переходу, безумовного переходу, арифметичні і логічні операції, команди пересилання даних, команди роботи з бітами. В останніх версіях кристалів AVR сімейства mega реалізована функція апаратного множення, що надає новим мікроконтролерам ще більшої привабливості з точки зору розробника.

За різноманітністю і кількістю реалізованих інструкцій AVR більше схожі на CISC-, ніж на RISC-процесори. Наприклад, у PIC-контролерів система команд налічує до 75 різних інструкцій, а у MCS51 (i8051) вона складає 111. У цілому прогресивна RISC-архітектура AVR у поєднанні з наявністю регістрового файлу і розширеної системи команд дозволяє в короткі терміни створювати працездатні програми з ефективним кодом як за компактністю реалізації, так і за швидкістю виконання.

4.5. ПРОГРАМУВАННЯ МІКРОКОНТРОЛЕРІВ

Значному поширенню мікроконтролерів неабиякою мірою сприяла можливість змінити функціональність, а іноді навіть весь принцип роботи пристрою вже після виготовлення плати і монтажу. Це можливо завдяки гнучкій архітектурі і можливості програмного задання алгоритму роботи схем (програмуванню). Для зміни коду програми не треба перепаявати деталі, різати доріжки, міняти конструкцію і топологію системи. Досить підключити кабель для програмування і відновити програму в пам'яті мікросхеми, що займає декілька секунд.

Програмування дозволяє сильно прискорити розробку нової системи, але, як завжди, якщо є переваги, є й недоліки. Для освоєння принципів програмування і написання програм треба мати уявлення про мови програмування Сі або асемблер. І хоча освоїти їх основи нескладно, усе ж таки це потребує зусиль. Також код програм для мікроконтролера повинен вміщуватися в доступну пам'ять, яка вже є в мікросхемі. Більшість мікроконтролерів не мають зовнішньої шини пам'яті, і тому використання зовнішніх мікросхем пам'яті для зберігання великих масивів даних зазвичай ускладнене і часто призводить до помітного здорожчання усієї системи.

Для програмування використовуються компілятори і середовища розробки, які перетворюють код програми на мовах високого рівня в компактний машинний код, ідентично з середовищами розробки програм для звичайного ПК. Важливі ділянки коду, які треба виконати максимально ефективно, нерідко пишуть на асемблері.

Виробники і постачальники мікроконтролерів часто пропонують вже перевірені бібліотеки і приклади коду для швидкого освоєння різних алгоритмів. Є навіть симулятори для деяких сімейств мікросхем, які дозволяють аналізувати і перевіряти роботу усієї системи і програми так, немов використовується справжній пристрій. Прикладами таких симуляторів можна назвати Microchip MPLAB для PIC, LabCenter Proteus. Симулятори дозволяють бачити стан внутрішніх регістрів, стан кожного виводу і реакцію системи на зовнішні події. У той же час програмна симуляція не дозволить перевірити роботу системи в складних умовах, з різними фізичними діями, тому для постійної роботи з мікроконтролерними системами більше виправдано використання відладчиків.

Відладчик – це пристрій зв'язку мікроконтролера з середовищем розробки на ПК, що дозволяє аналізувати стан усіх ключових вузлів мікроконтролера, редагувати вміст пам'яті і змінних, покроково виконувати програму, аж до повної зупинки програми. Для підключення відладчиків у більшості сучасних мікроконтролерів використовується інтерфейс JTAG.

У польових умовах також часто використовується можливість оновлення коду програми (так званої, «прошивки») за допомогою традиційних інтерфейсів RS232, USB, Ethernet. Це досягається завдяки використанню спеціальних завантажувачів (bootloader), що програмуються в мікроконтролерах.

4.6. ПЕРИФЕРІЙНІ ІНТЕРФЕЙСИ І МОЖЛИВОСТІ ЗВ'ЯЗКУ

Більшість мікроконтролерів використовуються в різних пристроях контролю і управління, коли необхідно вимірювати які-небудь зовнішні фізичні величини і сигнали. Для цього часто використовуються блоки АЦП (аналого-цифрові перетворювачі; англ. ADC). Мікропроцесори є цифровими схемами і можуть розпізнавати тільки двійковий код, тобто одиниці або нулі. Процесор не може сам виміряти рівень напруги на виході. Блок АЦП використовується для перетворення аналогових сигналів на двійковий код, зрозумілий процесору. Є також блоки зворотнього перетворення ЦАП (цифро-аналогові перетворювачі; англ. DAC), які залежно від вхідного двійкового коду видають певну напругу на виході мікроконтролера.

Окрім АЦП і ЦАП, більшість мікроконтролерів мають різні таймери. Таймер рахує кількість тактів до певного значення, заданого в програмі. Як тільки таймер досягає певного значення, видається сигнал на переривання і обробляється подія, що настала (виконується відповідна програма оброблення переривання). Таймери можуть бути різної довжини: 8-бітовий може зберігати будь-яке значення від 00000000 (00h) до 11111111 (FFh), 16-бітовий – від 0000h до FFFFh. Таймери зручно використовувати для підрахунку часу або виміру часових параметрів сигналів, а також для формування імпульсів із заданими затримками.

Наприклад, таймер працює з частотою 1 МГц і має довжину 8 бітів. Від 0 до максимального значення він дійде за $(1 \text{ біт} / 1000000 \text{ Гц}) \times 256 \text{ біт} = 256 \text{ мкс}$. 16-бітовий таймер із такою ж частотою дійшов би до максимального значення вже за 65536 мкс. Припустимо, що вирішується задача виміру температури 100 разів за секунду. Для вирішення задачі будемо використовувати 8-бітовий таймер, відлік якого лаштуємо до максимального значення. Кожне досягнення максимуму, або переповнення таймера, відбуватиметься через кожні 256 мікросекунд. Введемо змінну X для підрахунку кількості переривань від таймера. При кожному перериванні від таймера будемо збільшувати на одиницю змінну X .

Коли X досягне значення 39 (0,01 (потрібна нам затримка) / 0,000256 (максимальна затримка 8-бітового 1 МГц таймера) = 39,0625 переповнень), видаємо сигнал на початок виміру температури. Через похибку реальна частота вимірів складе 100,160256 Гц замість бажаних 100. Точного значення можна досягти, змінивши значення, до якого буде рахувати таймер, або опорну частоту відліку (у нашому випадку 1 МГц).

У мікроконтролерах часто є блоки широкоімпульсної модуляції (англ. *PWM*), які дозволяють управляти перетворювачами напруги, струмами напруги, двигунами без необхідності створення складних схем управління і монтажу та великої кількості додаткових деталей. Для невеликих потужностей (менше 200 Вт) зазвичай досить встановити оптосимістор.

Універсальний асинхронний прийомо-передатчик (англ. *UART*) дозволяє приймати і передавати дані за допомогою мікроконтролерів і всього декількох дротів у послідовному режимі на швидкостях до декількох мегабіт.

Нерідко мікроконтролери мають і інші інтерфейси зв'язку, наприклад I2C, SPI, CAN, Ethernet, MAC, SSC, I2S. Навіть у разі відсутності апаратного блока з потрібним інтерфейсом у разі потреби можна реалізувати його програмно за допомогою портів

введення/виведення загального призначення (GPIO або просто PIO). Оброблення стану і прочитування сигналів функціями портів відбувається значно повільніше, проте завдяки можливості реалізації повністю на програмному рівні є можливість практично реалізувати будь-який низькошвидкісний інтерфейс.

Існують і інші блоки, які використовуються в різних сімействах мікроконтролерів, наприклад шифратори, контролери пам'яті, рідкокристалічних екранів, пристроїв декодування. Проте саме АЦП, ЦАП, таймери, обробники переривань і інтерфейсні порти є тією базовою будівельною «цеглою», яка використовується практично в кожній мікроконтролерній системі.

Питання для самоперевірки

1. Наведіть приклади цифрових пристроїв, побудованих на базі мікроконтролерів.
2. Із яких основних елементів складається мікроконтролер?
3. Наведіть особливості мікроконтролерів як окремого виду МПС.
4. Які типи архітектур використовуються в мікроконтролерах?
5. Наведіть приклади сімейств мікроконтролерів і порівняйте їх за швидкістю виконання операцій.
6. Наведіть особливості архітектури MKS-51.
7. Наведіть позначення та особливості використання регістрів загального призначення в мікроконтролерах MKS-51.
8. Наведіть способи адресації операндів, які використовуються в архітектурі MKS-51.
9. На які групи можна розбити команди мікроконтролерів MKS-51?
10. Наведіть приклади команд пересилання даних із різними способами адресації.
11. Наведіть приклади арифметичних команд із різними способами адресації.
12. Наведіть приклади команд зсуву з різними способами адресації.
13. Наведіть приклади команд передання управління з різними способами адресації.
14. Наведіть приклади логічних команд із різними способами адресації.
15. Яку структуру має регістровий файл мікроконтролера типу AVR?
16. Як функціонують регістри мікроконтролера типу AVR?
17. Як відбувається програмування мікроконтролерів?
18. Які мови високого рівня використовуються для програмування мікроконтролерів?
19. Які периферійні інтерфейси використовуються в мікроконтролерних системах?

РОЗДІЛ 5

МІКРОПРОЦЕСОР

Мікропроцесор (англ. *microprocessor*) – інтегральна схема, яка виконує функції центрального процесора (ЦП) або спеціалізованого процесора. Сьогодні слово «мікропроцесор» є практично повним синонімом слова «процесор», оскільки функціональний блок, який на ранніх стадіях розвитку обчислювальної техніки представляв собою цілу плату чи навіть шафу, тепер вміщається в одну невеличку інтегральну схему із сотнями мільйонів транзисторів усередині. З середини 80-х рр. XX ст. мікропроцесори витіснили інші види ЦП. Винятком із цього правила сьогодні, мабуть, є тільки центральні процесорні пристрої деяких суперкомп'ютерів, які навіть у наш час є складними комплексами великих (ВІС) і надвеликих (НВІС) інтегральних схем.

Відомості, які наведені в цьому розділі, є теоретичним підґрунтям основ програмування на мові асемблера.

5.1. МІКРОПРОЦЕСОР ТИПУ INTEL

Досягнення фірми Intel у мистецтві проектування і виробництва напівпровідників зробили можливим виготовлення потужних процесорів усе менших розмірів. Розробники мікропроцесорів у даний час працюють із технологією, яка має назву «комплементарний технологічний процес Метал – Оксид – Напівпровідник (CMOS)» з розділом менше ніж мікрон. Використання субмікронних технологій дозволяє розробникам фірми Intel розташовувати більше транзисторів на кожній підкладці (шарі кристалу). Це зробило можливим збільшити кількість транзисторів для сімейства x86 від 29 тис. в Intel 8086 до 1,2 млн у процесорі Intel 486 DX2. Найвищим досягненням є процесор Pentium, який виконаний по 0,8-мікронній BiCMOS технології і містить 3,1 млн транзисторів. Технологія BiCMOS поєднує переваги двох технологій: біполярної (швидкість) і CMOS (мале енергоспоживання). За допомогою більше, ніж у два рази більшої кількості транзисторів Pentium процесора в порівнянні з Intel 486 розробники помістили на підкладці компоненти, які раніше розташовувались зовні процесора. Наявність компонентів усередині зменшує час доступу, що істотно збільшує продуктивність.

Сучасні представники сімейства Intel 80x86 є 32-розрядними процесорами. У нових моделях з'явилося 64-розрядне розширення. Історія 32-бітових процесорів Intel (архітектура IA-32) почалася з процесора 80386. Він увібрав у себе всі риси

своїх 16-бітових попередників 8086/88 і 80286 для забезпечення сумісності з величезним об'ємом програмного забезпечення, що існувало на момент його появи.

Потребує пояснення поняття розрядності адреси і розрядності даних. Розрядність адреси визначає кількість біт (16, 32 або 64) у регістрах, що формують адресу даних або команд, розташованих у пам'яті. Розрядність даних визначає, скільки біт використовується в командах, що оперують словами даних.

Кожному режиму роботи процесора відповідає своя розрядність (за умовчанням). При необхідності для кожної виконуваної команди (інструкції) розрядність адреси або/і операнда може змінюватися за допомогою спеціальних префіксів (байтів перед кодом операції). 32-бітові регістри процесорів дозволяють безпосередньо адресувати до 4 Гб пам'яті, що за часів появи процесора 80386 можна було вважати недосяжним. Вбудований блок управління пам'яттю підтримує механізми сегментації і сторінкової трансляції адрес.

64-розрядні розширення процесорів x86 у першу чергу призначені для радикального збільшення об'єму пам'яті, що адресується: 64-бітові регістри дозволяють адресувати до $2^{64} - 1$ байт. Це число і є межею об'єму віртуальної пам'яті 64-розрядного процесора, але поки використовують тільки молодші 48 біт адреси.

Процесори надають чотирирівневу систему привілеїв для захисту пам'яті, введення/виведення і переривань, а також механізм перемикавання завдань для багатозадачних ОС. Система команд процесорів постійно розширюється при збереженні всіх команд попередніх процесорів x86. З розширенням системи команд розширюється і набір архітектурних регістрів (MMX, XMM, нові загальні 64-бітові регістри).

Процесори можуть працювати в різних режимах, що визначають можливості адресації пам'яті і захисту: у реальному (16-розрядному) режимі процесора 8086, у режимі віртуального процесора 8086 (V86), у захищеному 32-розрядному (і захищеному 16-розрядному) режимах. Режим роботи процесора задається операційною системою з урахуванням режиму роботи програм (задач, додатків). У процесорів з 64-розрядним розширенням з'являються нові режими, серед яких є і режими, що забезпечують сумісність із 32-розрядними операційними системами і задачами. Нові режими використовуються тільки в 64-розрядних ОС, а повністю їх переваги доступні тільки 64-розрядним задачам.

Режими роботи процесорів

32-розрядні процесори можуть працювати в одному з наступних режимів:

– *режим реальної адресації* (real address mode), або просто *реальний режим* (real mode), повністю сумісний з 8086. У цьому режимі можлива адресація до 1 Мб фізичної пам'яті (насправді, як і у 80286, майже на 64 Кб більше);

– *захищений режим віртуальної адресації* (protected virtual address mode), або просто *захищений режим* (protected mode). У цьому режимі у процесора включаються механізми сегментації і сторінкової адресації. Механізм сегментації дозволяє підтримувати віртуальну пам'ять об'ємом до 64 Тбайт. На практиці використовується тільки сторінкова адресація, завдяки якій кожній задачі надається до 4 Гб віртуального адресного простору. За умовчанням і адреси, і операнди мають розрядність 32 біти. У захищеному режимі процесор може виконувати додаткові інструкції, недоступні в реальному режимі; ряд інструкцій, пов'язаних із переданням управління, обробленням переривань. Деякі інструкції виконуються інакше, ніж у реальному режимі;

– *режим віртуального процесора 8086* (virtual 8086 mode, V86) є особливим станом захищеного режиму, у якому процесор функціонує як 8086 (16-бітові адреса і дані). На одному процесорі в такому режимі можуть паралельно виконуватися декілька задач з ізолюваними один від одного ресурсами. При цьому використання фізичного адресного простору пам'яті управляється механізмами сегментації і трансляції сторінок. Спроби виконання неприпустимих команд, виходу за рамки відведеного простору пам'яті і дозволеної області введення/виведення контролюються системою захисту;

– *розширений режим віртуального процесора 8086* (enhanced virtual 8086 mode, EV86) є більш ефективним за рахунок віртуалізації та оптимізації переривань;

– «*нереальний*» режим (unreal mode, він же big real mode) – це «неофіційний» режим, який підтримують усі 32-розрядні процесори. Він дозволяє адресуватися до 4 Гб простору пам'яті. У цьому режимі інструкції виконуються так само, як і в реальному режимі, але за допомогою додаткових сегментних регістрів FS і GS програми мають безпосередній доступ до даних у всій фізичній пам'яті;

– у *режимі системного управління* (system management mode – SMM) процесор виходить в інший, ізолюваний від решти режимів простір пам'яті. Цей режим використовується зі службовими і налагоджувальними цілями. З його допомогою, наприклад, виконуються функції управління енергоспоживанням, емулюються звернення до неіснуючих апаратних засобів (емуляція клавіатури і миші PS/2 для USB).

Для процесорів x86-64 вищеперелічені режими є «легальними» (legacy mode). Крім того, з'явився новий режим long mode з двома підрежимами:

- 64-розрядний режим (64-bit mode) – це режим повної підтримки 64-бітової віртуальної адресації і 64-бітових розширень регістрів. У цьому режимі використовується загальний сегмент для коду програми, даних і стека. За умовчанням розрядність адреси складає 64 біти, а операндів (для більшості інструкцій) – 32 біти, проте за допомогою певного префікса (REX) можна заказати 64-бітові операнди;

- *режим сумісності* (compatibility mode) дозволяє 64-розрядним ОС працювати з 32- і 16-розрядними додатками. Для додатків процесор виглядає як звичайний 32-розрядний зі всіма атрибутами захищеного режиму, сегментацією і сторінковою трансляцією. 64-розрядні властивості використовуються тільки операційною системою, що відбивається в процедурах трансляції адрес, оброблення винятків і переривань. Режим включається операційною системою для конкретної задачі.

32-розрядні ОС використовують процесори x86-64 тільки в режимі legacy mode (як звичайний процесор IA-32).

5.2. ПРОГРАМНА МОДЕЛЬ ПРОЦЕСОРІВ ТИПУ IA-32

Виконання того чи іншого алгоритму можливе за наявності мікропроцесора та пристроїв, у яких зберігається програма. Відомо, що програма – це сукупність команд (правил), що виконуються в послідовності, заданій алгоритмом. Команди вибираються з пам'яті в послідовності, яку визначає процесор. Процесор визначає адреси елементів пам'яті, у яких зберігаються необхідні дані. Дані передаються в процесор, де перетворюються у відповідності з командами, і результати операції передаються знову в пам'ять.

Будь-якій програмі на асемблері для її виконання виділяється певний набір ресурсів комп'ютера, більшість із яких надається процесором. Вони потрібні програмі для оброблення і зберігання в пам'яті команд і даних, а також для отримання інформації про власний поточний стан і стан процесора.

Процесори можуть оперувати з операндами різних типів і розмірів:

- цілими числами (зі знаком і без знаку) розміром у байт, словом (16 біт), подвійним словом (DWord, 32 біти), збільшеним учетверо словом (QWord, 64 біти) і подвійним збільшеним учетверо словом (DQWord, 128 біт);

- рядками байтів, слів, подвійних і збільшених учетверо слів;
- бітами, бітовими полями і рядками бітів;

– числами у форматі з рухомою крапкою (FP) розміром у 32, 64 і 80 біт.

Можливість роботи з довгими операндами (64 і 128 біти) з'явилася в процесорах з розширеннями MMX і XMM, базова архітектура IA-32 таких можливостей не надавала. Операнди інструкцій можуть знаходитися в регістрах процесора, пам'яті (або в порту введення/виведення), а також у самій інструкції (безпосередній операнд). Найефективніше процесор працює з операндами, розташованими в його регістрах.

Програмну модель процесора в архітектурі IA-32 (intel architecture) процесорів intel складає наступний набір ресурсів (див. рис. 5.1):

- простір пам'яті, що адресується, до 2^{32} байт (4 Гб), для Pentium II і вище – до 2^{36} байт (64 Гб);
- регістри для зберігання даних загального призначення;
- сегментні регістри;
- регістри стану і управління;
- регістри пристрою для обчислень із рухомою крапкою (співпроцесора);
- набір регістрів MMX-розширення для цілих чисел, відображених на регістрі співпроцесора (які вперше з'явилися в архітектурі процесора Pentium MMX);
- набір регістрів MMX-розширення з рухомою крапкою (які вперше з'явилися в архітектурі процесора Pentium III);
- програмний стек – спеціальна інформаційна структура, для роботи з якою передбачені відповідні машинні команди.

Крім того, до ресурсів, які підтримуються архітектурою IA-32, необхідно віднести порти введення/виведення, лічильники моніторингу продуктивності.

Програмні моделі більш ранніх процесорів (i486, перші Pentium до Pentium MMX включно) відрізняються меншим розміром простору оперативної пам'яті (максимально 2^{32} адреси, оскільки розрядність їх шини адреси складає 32 біти), і відсутністю деяких груп регістрів. Тому на рис. 5.1 для кожної групи регістрів у дужках показано, починаючи з якої моделі вона з'явилася в програмній моделі процесорів intel. Якщо такого позначення немає, то це означає, що ця група регістрів була присутня в процесорах i386 і i486. Що стосується ще більш ранніх процесорів i8086/88, то насправді вони теж повністю представлені на схемі, але складають лише невелику її частину. У програмну модель цих процесорів входять 8- і 16-розрядні регістри загального призначення, сегментні регістри, регістри FLAGS, IP і адресний простір пам'яті розміром до 1 Мб.

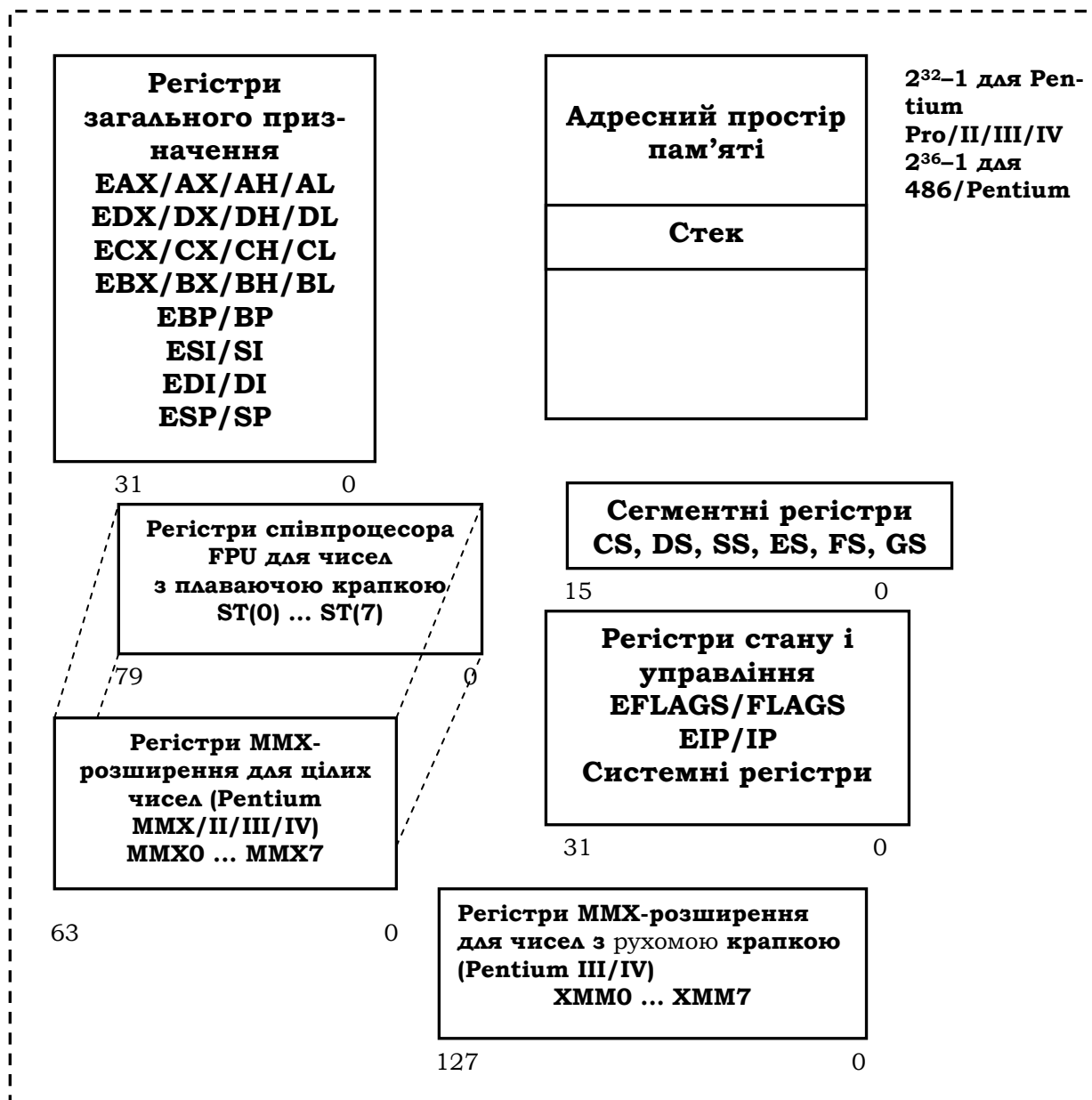


Рис. 5.1. Програмна модель процесорів IA-32

Набір реєстрів

Регістрами називають елементи високошвидкісної пам'яті, які розташовані усередині процесора в безпосередній близькості від його ALU. Доступ до них здійснюється незрівнянно швидше, ніж до елементів оперативної пам'яті. Більшість реєстрів мають певне функціональне призначення. З точки зору програміста, їх можна розділити на дві великі групи.

Першу групу утворюють призначені для користувача реєстри, до яких відносяться:

– реєстри загального призначення EAX/AX/AH/AL, EBX/BX/BH/BL, EDX/DX/DH/DL, ECX/CX/CH/CL, EBP/BP, ESI/SI, EDI/DI, ESP/SP, призначені для зберігання даних і ад-

рес, програміст може їх використовувати (з певними обмеженнями) для реалізації своїх алгоритмів*.

- сегментні реєстри CS, DS, SS, ES, FS, GS, які слугують для зберігання значень, інтерпретація яких залежить від режиму роботи процесора. У реальному режимі сегментні реєстри містять адресу параграфа початку сегмента в пам'яті. У захищеному режимі сегментні реєстри зберігають індекс входу в одну із системних таблиць дескрипторів – глобальну GDT або локальну LDT;

- реєстри співпроцесора ST(0), ST(1), ST(2), ST(3), ST(4), ST(5), ST(6), ST(7), що призначені для написання програм, котрі оперують даними з рухомою комою;

- реєстри цілочисельного MMX-розширення MMX0, MMX1, MMX2, MMX3, MMX4, MMX5, MMX6, MMX7;

- реєстри XMM-розширення з рухомою комою XMM0, XMM1, XMM2, XMM3, XMM4, XMM5, XMM6, XMM7;

- реєстри стану і управління (реєстр прапорів EFLAGS/FLAGS і реєстр покажчика команди EIP/IP), які містять інформацію про стан процесора, виконуваної програми і дозволяють змінити цей стан.

До другої групи входять системні реєстри, які забезпечують різні режими роботи процесора, обслуговують сервісні функції, а також реєстри, специфічні для певної моделі процесора.

До системних реєстрів процесорів IA-32 відносяться:

- реєстри CR0–CR4, які визначають режим роботи процесора і характеристики поточного завдання;

- реєстри управління пам'яттю GDTR, IDTR LDTR і TR, які використовуються в захищеному режимі роботи процесора для локалізації структур цього режиму;

- реєстри DR0–DR7, що призначені для контролю і управління різними аспектами відладки;

- реєстри типів областей пам'яті MTRR, що використовуються для апаратного управління кешуванням з метою призначення відповідних властивостей областям пам'яті;

* У позначеннях багатьох реєстрів програмної моделі присутня розділова похила риска. Це не різні реєстри, а частини одного великого 32-розрядного реєстра. Похила риска означає, що ці реєстри можна використовувати в програмі як окремі об'єкти. Так зроблено для забезпечення працездатності програм, написаних для колишніх 16-розрядних моделей процесорів фірми Intel, починаючи з i8086. Процесори i486 і Pentium включають в основному 32-розрядні реєстри. Їх кількість, за винятком сегментних реєстрів, така ж, як і у i8086, але місткість більша, що і відбито в позначеннях, – вони мають приставку E (Extended);

– машинно-залежні регістри MSR, які задіюються для управління процесором, контролю за його продуктивністю та отримання інформації про помилки.

Регістри загального призначення

Регістри загального призначення використовуються для зберігання операндів команд для цілочисельної арифметики, а також адрес і компонентів адрес (див. рис. 5.2).

		AX	
EAX		AH	AL
		DX	
EDX		DH	DL
		CX	
ECX		CH	CL
		BX	
EBX		BH	BL
EBP		BP	
ESI		SI	
EDI		DI	
ESP		SP	
	31	15	0

Рис. 5.2. Регістри загального призначення

На зберігання операндів не накладаються особливі обмеження, але в певних умовах деякі з регістрів мають жорстке функціональне призначення, закріплене на рівні логіки роботи машинних команд.

У зв'язку з цим серед усіх регістрів загального призначення особливо слід зазначити регістр ESP. У ньому зберігається покажчик вершини стека програми, тому зберігати в ньому якінебудь операнди безрозсудно і небезпечно.

Перерахуємо регістри, що відносяться до групи регістрів загального призначення і фізично знаходяться в процесорі усередині арифметико-логічного пристрою (рис. 5.2):

– регістр-акумулятор (*accumulator register*) EAX/AX/AH/AL, що використовується для зберігання проміжних даних, результатів операцій. У деяких командах його використання є обов'язковим;

– базовий регістр (*base register*) EBX/BX/BH/BL, який стандартно використовується для зберігання базової адреси певного об'єкта в пам'яті;

– регістр-лічильник (*count register*) ECX/CX/CH/CL, котрий застосовується в командах, що виконують багатократні дії;

– реєстр даних (*data register*) EDX/DX/DH/DL так само, як і реєстр EAX/AX/AH/AL, зберігає проміжні дані (у деяких командах його явне використання обов'язкове, в інших – його участь мається на увазі неявно).

Наступні два реєстри призначені для підтримки так званих ланцюжкових операцій, тобто операцій, що здійснюють послідовне оброблення ланцюжків елементів, кожен із яких може мати довжину 32, 16 або 8 бітів.

До них відносяться:

– реєстр індексу джерела (*source index register*) ESI/SI, який у ланцюжкових операціях містить поточну адресу елемента в ланцюжку-джерелі;

– реєстр індексу приймача (*destination index register*) EDI/DI, що в ланцюжкових операціях містить поточну адресу елемента в ланцюжку-приймачі.

Для роботи зі стеком у системі команд процесора є спеціальні команди, а в програмній моделі процесора для цього існують спеціальні реєстри:

– реєстр покажчика стека (*stack pointer register*) ESP/SP, який містить покажчик на вершину стека в поточному сегменті стека;

– реєстр покажчика бази точки кадра стека (*base pointer register*) EBP/BP, що призначений для організації довільного доступу до даних усередині стека.

Сегментні реєстри

Процесори Intel апаратно підтримують сегментну організацію програми. Це означає, що будь-яка програма складається як мінімум із трьох сегментів: коду, даних і стека. Логічно машинні команди в архітектурі IA-32 побудовані так, що при вибірці кожної команди для доступу до даних програми або до стека неявно використовується інформація з цілком певних сегментних реєстрів. Залежно від режиму роботи процесора за їх вмістом визначаються адреси пам'яті, із яких починаються відповідні сегменти. У програмній моделі IA-32 є шість сегментних реєстрів CS, SS, DS, ES, GS, FS, які слугують для доступу до чотирьох типів сегментів (див. рис. 5.3).

CS
DS
SS
ES
FS
GS

Рис. 5.3. Сегментні реєстри

Сегмент коду містить команди програми. Для доступу до цього сегмента слугує регістр сегмента коду (*code segment register*) CS. Він містить значення, яке по-різному інтерпретується процесором залежно від поточного режиму роботи. У реальному режимі – це адреса першого параграфу сегмента, у захищеному – індекс елементу в таблиці дескрипторів (GDT або LDT).

Сегмент даних зберігає оброблювані програмою дані. Для доступу до цього сегмента служить регістр сегмента даних (*data segment register*) DS, у якому зберігається адреса сегмента даних поточної програми.

Сегмент стека обслуговує область пам'яті, яку називають стеком. Роботу зі стеком процесор організовує за наступним принципом: останній записаний у цю область елемент вибирається з неї першим. Доступ до області стека виконується через регістр сегмента стека (*stack segment register*) SS, який містить адресу сегмента стека.

Додатковий сегмент даних використовується в командах у випадку потреби. Неявно алгоритми виконання більшості машинних команд припускають, що оброблювані дані розташовані в сегменті даних, адреса якого знаходиться в регістрі сегмента даних DS. Якщо програмі недостатньо одного сегмента даних, то вона має можливість одночасно задіювати ще три додаткові сегменти даних, адреси яких повинні міститися в регістрах додаткового сегмента даних (*extension data segment registers*) ES, GS, FS. Серед них слід особливо виділити регістр ES. Він використовується для забезпечення ланцюжкових команд процесора, адресуючи собою операнд-джерело.

Регістри стану і управління

У процесор Intel включено два регістри (див. рис. 5.1), що постійно містять інформацію про стан як самого процесора, так і програми, команди якої він у даний момент обробляє. Це регістр показчика команди EIP/IP та регістр прапорів EFLAGS/FLAGS. За допомогою цих регістрів можна також певним чином управляти станом процесора.

Регістр показчика команд (*instruction pointer register*) EIP/IP має розрядність 32/16 біт і містить зміщення адреси команди відносно початку сегмента коду, яка підлягає виконанню наступною. Цей регістр безпосередньо недоступний програмісту, тобто його не можна вказувати як операнд команди. Але до нього можливий непрямої доступ, такий як завантаження і зміна його вмісту в результаті роботи команд управління програмним потоком. До таких команд відносяться команди умовних і безумовних переходів, виклику процедур і повернення з процедур. Виникнення переривань також призводить до модифікації регістра EIP/IP.

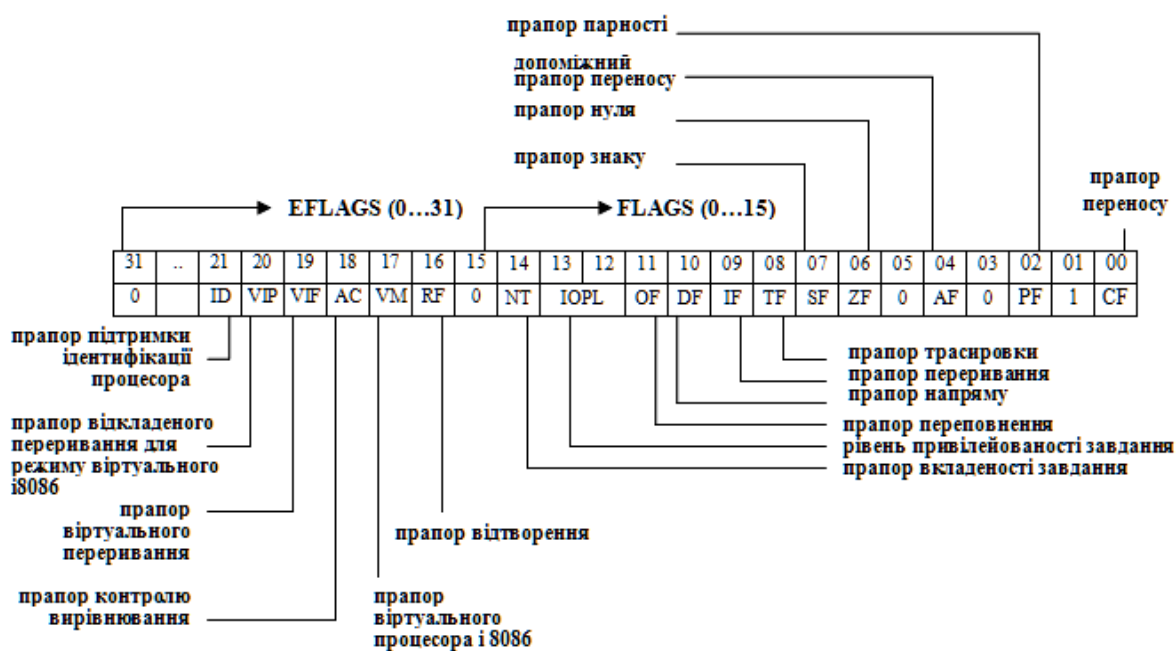


Рис. 5.4. Регістр EFLAGS

Розрядність *регістра прапорів* (FLAG REGISTER) EFLAGS/FLAGS дорівнює 32/16 біт. Окремі біти цього регістра мають функціональне призначення і називаються прапорами. Молодша частина регістра EFLAGS/FLAGS повністю аналогічна регістру FLAGS процесора i8086. На рис. 5.4 показаний вміст регістра EFLAGS.

Виходячи з особливостей використання прапори регістра EFLAGS/FLAGS діляться на три групи: прапори стану, системні прапори і прапор управління.

До *прапорів стану* відносяться: прапор переносу CF, прапор парності PF, допоміжний прапор переносу AF, прапор нуля ZF, прапор знаку SF, прапор переповнення OF, поле рівня привілейованості завдання IOPL, прапор вкладеності завдання NT (див. рис. 5.4).

Прапори стану відбивають особливості результату виконання арифметичних або логічних операцій і можуть змінюватися свідомо.

Прапор переносу (carry flag) CF (біт 0) дорівнює 1, якщо арифметична операція призвела до переносу із старшого біта результату або в старший біт результату. Перенос може бути ініційований командою зсуву. Старшим є 7-й, 15-й або 31-й біт залежно від розмірності операнда. Якщо CF дорівнює 0, то переносу не було. В алгоритмах беззнакової арифметики цей прапор використовується як показчик переповнення.

Прапор парності (parity flag) PF (біт 2) дорівнює 1, якщо 8 молодших розрядів (цей прапор дійсний тільки для 8 молодших розрядів операнда будь-якого розміру) результату містять

парне число одиниць. Якщо PF дорівнює 0, то серед 8-ми молодших розрядів результату кількість одиниць непарна.

Допоміжний прапор переносу (auxiliary carry flag) AF (біт 4) застосовується тільки для команд, що працюють із числами певних форматів. Цей прапор фіксує факт позики з молодшої тетради результату: якщо він дорівнює 1 – у результаті операції складання був проведений перенос із 3-го розряду в старший або при відніманні була позика в 3-й розряд молодшої тетради зі старшої тетради; якщо він дорівнює 0 – переносу або позики в 3-й розряд (або з 3-го розряду) молодшої тетради результату не було. Цей прапор використовується у двійково-кодованій десятковій арифметиці (BCD – Binary-Coded Decimal).

Прапор нуля (zero flag) ZF (біт 6) дорівнює 1, якщо результат нульовий. При ZF = 0 результат ненульовий.

Прапор знаку (sign flag) SF (біт 7) відбиває стан старшого біта результату (біти 7, 15 або 31 для 8-, 16- або 32-розрядних операндів відповідно): якщо SF = 1 – старший біт результату дорівнює 1; якщо SF = 0 – старший біт результату також 0.

Прапор переповнення (overflow flag) OF (біт 11) використовується для фіксації факту втрати значущого біта при арифметичних операціях над даними зі знаком: якщо OF = 1, то в результаті операції відбувся перенос у старший знаковий біт результату або позика зі старшого знакового біта результату (біти 7, 15 або 31 для 8-, 16- або 32-розрядних операндів відповідно); нульове значення прапора OF говорить про відсутність переносу або позики.

Прапор напряму (directory flag) DF (біт 10) використовується ланцюжковими командами та визначає напрям поелементного оброблення даних цими командами. Якщо DF = 0, то напрям оброблення даних – від початку ланцюжка до його кінця, тобто від молодших адрес до старших. Якщо DF = 1, то напрям – від кінця ланцюжка до його початку, тобто від старших адрес до молодших. Стан прапора DF можна змінити командами CLD (зняти прапор DF) і STD (встановити прапор DF).

Із прапорів стану тільки CF можна змінювати безпосередньо (командами STC, CLC і CMC). Прапор CF зберігає результат виконання команд BT, BTS, BTR, BTC.

Прапори стану використовуються командами знакової, беззнакової та BCD-арифметики цілих чисел. Для фіксації переповнення в алгоритмах знакової арифметики використовують прапор OF, в алгоритмах беззнакової арифметики – прапор CF, для BCD-арифметики – прапор AF.

Команди умовних переходів використовують один або декілька прапорів стану для перевірки умов переходу.

До *системних прапорів* відносяться: прапор трасування TF, прапор переривання IF, прапор відтворення RF, прапор віртуального процесора i8086, прапор контролю вирівнювання AC, прапор віртуального переривання VIF, прапор відкладеного переривання для режиму віртуального i8086, прапор підтримки ідентифікації процесора ID.

Ці прапори використовуються на системному рівні; припускається, що прикладні програми не можуть змінювати їх значення.

Системні прапори і поле IOPL управляють введенням/виведенням, маскованими перериваннями, відладкою, перемиканням між завданнями і режимом віртуального процесора i8086.

Прапор трасування (trace flag) TF (біт 8) призначений для організації покрокової роботи процесора: якщо він дорівнює 1, процесор генерує переривання з номером 1 після виконання кожної машинної команди (може використовуватися при відладці програм); якщо він дорівнює 0, робота в звичайному режимі.

Прапор переривання (interrupt enable flag) IF (біт 9) призначений для дозволу або заборони (маскування) апаратних переривань (переривань по входу INTR): коли він дорівнює 1, апаратні переривання дозволені; якщо він дорівнює 0, апаратні переривання заборонені.

Поле (input/output privilege level) IOPL (біти 12, 13) визначає рівень привілейованості операції введення/виведення. IOPL використовується в захищеному режимі роботи процесора для контролю доступу до команд введення/виведення залежно від призначеного завданню пріоритету.

Прапор вкладеності завдання (nested task) NT (біт 14) використовується в захищеному режимі роботи процесора для запам'ятовування того факту, що одне завдання пов'язане з іншим по ланцюжку викликів. Робота ланцюжка завдань, пов'язаних між собою, організується за допомогою команд CALL, IRET, регістра завдань TR і сегмента TSS.

Прапор відновлення (resume flag) RF (біт 16) застосовується при обробленні переривань від регістрів відладки.

Прапор режиму віртуального процесора i8086 (virtual 8086 mode) VM (біт 17) зберігає ознаку поточного режиму процесора: якщо він дорівнює 1, процесор функціонує в режимі віртуального процесора i8086; якщо він дорівнює 0, це означає роботу в реальному або захищеному режимі.

Прапор контролю вирівнювання (alignment check) AC (біт 18) призначений для дозволу контролю вирівнювання при зверненнях до пам'яті. Використовується спільно з бітом AM у

системному реєстрі CR0. Процесори IA-32 (починаючи з i80486) дозволяють розміщувати команди і дані з будь-якої адреси. Встановлення AM і CR0 в одиницю вимагає від процесора перевіряти вирівнювання при зверненні до пам'яті і у випадку відсутності вирівнювання генерувати виняткову ситуацію.

Прапор віртуального переривання (virtual interrupt flag) VIF (біт 19) використовується процесором Pentium за певних умов, одна з яких – робота процесора в режимі віртуального i8086. VIF є аналогом прапора IF. Прапор VIF використовується спільно з прапором VIP (біт 20).

Прапор відкладеного віртуального переривання (virtual interrupt pending flag) VIP (біт 20, тільки Pentium) встановлюється в одиницю для індикації відкладеного переривання. Застосовується при роботі в режимі віртуального процесора i8086 спільно з прапором VIF. Може бути встановлений тільки програмно, процесор лише читає його.

Прапор ідентифікації (identification flag) ID (біт 21) використовується для того, щоб показати факт підтримки процесором команди CPUID. Якщо програма може встановити або скинути цей прапор, то це означає, що ця модель процесора підтримує інструкцію CPUID. CPUID (CPU Identification) – це інструкція процесора для отримання інформації (ідентифікації) про процесор. Програмно можна визначити тип CPU та його можливості. Інструкція CPUID уперше з'явилась у процесорах i486.

Системні реєстри процесора

Сама назва говорить про те, що ці реєстри виконують специфічні функції в системі. Використання їх жорстко регламентоване. Саме вони забезпечують роботу захищеного режиму. Їх також можна розглядати як частину архітектури процесора, яка навмисно залишена в описі архітектури для того, щоб дозволити кваліфікованому системному програмістові виконати найбільш низькорівневі операції.

Системні реєстри можна розділити на три групи:

- 1) реєстри управління;
- 2) реєстри системних адрес;
- 3) реєстри відладки.

До складу системних реєстрів процесорів, починаючи з Pentium, введені наступні зміни: задіяний раніше зарезервованний реєстр управління CR4; додана група MSR-реєстрів (MSR – model specific register, модельно-залежні реєстри процесора), призначення і можливості яких залежать від архітектурі конкретної моделі процесора. Раніше їх функції частково виконували тестові реєстри, що увійшли тепер до складу MSR-групи. Для доступу до цих реєстрів передбачені спеціальні команди.

Регістри управління (Control Registers)

До групи реєстрів управління входять п'ять реєстрів: CR0, CR1, CR2, CR3, CR4. Вони призначені для загального управління системою. Регістри управління доступні тільки програмам із рівнем привілеїв 0 (правами цього сегмента відносно інших сегментів).

Фактично доступними є чотири реєстри (CR0, CR2, CR3, CR4). Регістр CR1, функції якого поки що не визначені, зарезервований для майбутнього застосування.

Регістр CR0 містить системні прапори роботи процесора, що управляють режимами і відбивають його стан глобально, незалежно від конкретних завдань, що виконуються. Він використовується при сторінковій моделі організації пам'яті.

Призначення кількох системних прапорів реєстра CR0:

- PE (protect enable), біт 0 – включає захищений режим роботи процесора. Стан цього прапора вказує на один з двох режимів: або реальний (PE = 0), або захищений (PE = 1);

- MP (math present), біт 1 – сигналізує про наявність співпроцесора. Завжди дорівнює 1;

- TS (task switched), біт 3 – слідує за моментом перемикавання завдань. Процесор автоматично встановлює цей прапор в 1 при перемиканні на виконання іншого завдання;

- AM (alignment mask), біт 18 – маска вирівнювання. Цей біт дозволяє (AM = 1) чи забороняє (AM = 0) контроль вирівнювання;

- CD (cache disable), біт 30 – здійснює блокування кеш-пам'яті. За допомогою цього біта можна заборонити (CD = 1) або дозволити (CD = 0) використання внутрішньої кеш-пам'яті (першого рівня);

- PG (paging), біт 31 – надає дозвіл (PG = 1) або забороняє (PG = 0) посторінкове перетворення.

Регістр CR2 використовується при сторінковій організації оперативної пам'яті для реєстрації ситуації, коли поточна команда звернулася за адресою, що належить сторінці, яка відсутня в даний момент часу в пам'яті. У такій ситуації в процесорі виникає виняткова ситуація номер 14, і лінійна 32-бітова адреса команди, що спричинила переривання, записується в реєстр CR2. Маючи цю інформацію, програма оброблення переривання номер 14 визначає потрібну сторінку, здійснює її завантаження в пам'ять і поновлює нормальну роботу програми.

Регістр CR3 також використовується при сторінковій організації пам'яті. Це так званий реєстр каталогу сторінок першого рівня. Він містить 20-бітову фізичну базову адресу каталогу сторінок поточного завдання. Цей каталог містить 10^{24} 32-бітових дескриптора, кожен із яких зберігає адресу таблиці

сторінок другого рівня. У свою чергу, кожна з таблиць сторінок другого рівня містить 10^{24} 32-бітових дескриптора, що адресують сторінкові кадри в пам'яті. Розмір сторінкового кадру – 4 Кб.

Регістр CR4 є сукупністю ознак в основному дозвільного характеру, які характеризують ті або інші архітектурні елементи, що вперше з'явилися в різних моделях процесорів Pentium. У якості прикладів таких властивостей можна навести наступні: підтримка 36-розрядної адресації, використання відкладених переривань у режимі віртуального процесора i8086, підтримка сторінок розміром 4 Мб і т. ін. Встановлюючи в регістрі CR4 ті або інші біти, можна включати або відключати підтримку цих властивостей.

Регістри системних адрес

Регістри системних адрес ще називають регістрами управління пам'яттю. Вони призначені для захисту програм і даних у мультизадачному режимі роботи процесора. При роботі в захищеному режимі процесора адресний простір розділяється на:

- глобальний – загальний для всіх завдань;
- локальний – окремий для кожного завдання.

Цим розподілом і пояснюється те, що в архітектурі процесора присутні наступні регістри системних адрес:

- регістр таблиці глобальних дескрипторів GDTR (*global descriptor table register*) має розмір 48 бітів і містить 32-бітну (біти 16–47) базову адресу глобальної таблиці дескрипторів (GDT) і 16-бітне (біти 0–15) значення граничного розміру, що є розміром у байтах таблиці GDT;

- регістр таблиці локальних дескрипторів LDTR (*local descriptor table register*) має розмір 16 біт і містить так званий селектор дескриптора локальної таблиці дескрипторів. Цей селектор є показником у таблиці GDT, який і описує сегмент, у якому знаходиться локальна таблиця дескрипторів (LDT);

- регістр таблиці дескрипторів переривань IDTR (*interrupt descriptor table register*) має розмір 48 біт і містить 32-бітову (біти 16–47) базову адресу таблиці дескрипторів переривань (IDT) і 16-бітне (біти 0–15) значення межі, що є розміром у байтах таблиці IDT;

- 16-бітовий регістр завдання TR (*task register*), що подібно до регістра LDTR містить селектор, тобто показник на дескриптор у таблиці GDT. Цей дескриптор описує поточний сегмент стану завдання (TSS – *task segment status*). TSS створюється для кожного завдання в системі, має жорстко регламентовану структуру і зберігає контекст (поточний стан) завдання. Основне призначення сегментів TSS – запам'ятовувати поточний стан завдання у момент перемикання на інше завдання.

Регістри відладки

Дуже цікава група реєстрів, призначених для апаратного контролю виконання програми. Засоби апаратної відладки вперше з'явилися в моделі i486. Процесор містить вісім реєстрів відладки, але реально з них використовуються тільки шість. Регістри DR0, DR1, DR2, DR3 мають розрядність 32 біти і призначені для завдання лінійних адрес чотирьох точок переривань. При цьому діє наступний механізм: будь-яка сформована поточною програмою адреса порівнюється з адресами в реєстрах DR0 ... DR3, і при збігу генерується виключення відладки з номером 1.

Регістр DR6 називається реєстром стану відладки. Біти цього реєстра встановлюються відповідно до причин, які викликали виникнення останнього переривання з номером 1. Перерахуємо ці біти та їх призначення:

- B0 – якщо встановлений в 1, то останнє виключення (переривання) виникло в результаті досягнення контрольної точки, зазначеної в реєстрі DR0;

- B1 – аналогічно B0, але для контрольної точки в реєстрі DR1;

- B2 – аналогічно B0, але для контрольної точки в реєстрі DR2;

- B3 – аналогічно B0, але для контрольної точки в реєстрі DR3;

- BD (біт 13) служить для захисту реєстрів відладки;

- BS (біт 14) встановлюється в 1, якщо виключення з номером 1 було викликано станом прапора TF = 1 в реєстрі EFLAGS;

- BT (біт 15) встановлюється в 1, якщо виключення з номером 1 було викликано перемиканням на завдання зі встановленим бітом T = 1 в TSS. Усі інші біти в цьому реєстрі заповнюються нулями. Обробник виключення з номером 1 за вмістом DR6 повинен визначити причину, за якої сталося виключення, і виконувати необхідні дії.

Регістр DR7 називається реєстром управління відладкою. У ньому для кожного з чотирьох реєстрів контрольних точок відладки відводяться поля, за допомогою яких можна уточнити умови, при яких слід згенерувати переривання: місце реєстрації контрольної точки знаходиться тільки в поточному завданні або в будь-якому завданні.

Відповідні ознаки займають молодші вісім бітів реєстра DR7: по два біти на кожен контрольну точку (фактично точку переривання), що задається реєстрами DR0, DR1, DR2, DR3. Перший біт із кожної пари – це так званий локальний дозвіл, його встановлення говорить про те, що точка переривання діє,

якщо вона знаходиться в межах адресного простору поточного завдання. Другий біт у кожній парі визначає глобальний дозвіл, тобто ця контрольна точка діє в межах адресних просторів усіх завдань, що обслуговуються в системі; тип доступу, по якому ініціюється переривання (тільки при вибірці команди, при записі або при записі/читанні даних). Біти, що визначають природу переривання, зосереджені в старшій частині цього регістра.

Більшість із системних регістрів програмно доступні. Не всі з них використовуються програмістами на практиці, але стислий опис, який ми надали в цьому розділі, має за мету викликати зацікавленість читача до подальшого дослідження архітектури процесора.

Блоки регістрів співпроцесора, MMX-, XMM-розширень

Блоки регістрів математичного співпроцесора FPU (*Floating Point Unit*), MMX- і XMM-розширень в архітектурі процесорів IA-32 займають окреме місце. Вони присутні не у всіх процесорах і є добудовами до центрального процесора з його набором звичайних цілочисельних регістрів. Ці блоки призначені для прискорення обчислень із даними певних форматів.

Нагадаємо формати даних, якими оперує процесор:

– цілі числа (зі знаком та без нього) розміром у байт; слова довжиною 16 бітів; подвійні слова довжиною 32 біти; слова довжиною 64 біти та 128 бітів; строчки байтів та слів будь-якої довжини; біти, бітові поля та строчки бітів; числа у форматі з рухомою крапкою (*FP – Floating Point*) розміром 32, 64 та 80 бітів.

Блок FPU є присутнім у всіх сучасних процесорах; його не було в i486SX, а в попередніх процесорах він як математичний співпроцесор встановлювався додатково. У різних поколіннях процесорів він називався по-різному: FPU (*Floating Point Unit* – блок чисел із рухомою крапкою) або NPX (*Numeric Processor extension* – числове розширення процесора). Блок FPU призначений для розширення обчислювальних можливостей центрального процесора – виконання арифметичних операцій, обчислення основних математичних функцій (тригонометричних, експоненти, логарифма) і т. ін.

Блок FPU містить стек із восьми 80-бітних регістрів і кілька допоміжних регістрів. Співпроцесор підтримує сім типів даних: 16-, 32-, 64-бітні цілі числа; 32-, 64-, 80-бітні числа із рухомою крапкою (FP-формати) і 18-бітні числа у двійково-десятковому (BCD) форматі. Застосування співпроцесора підвищує продуктивність обчислень у сотні раз.

Технологія MMX орієнтована на додатки мультимедіа, 2D/3D-графіку і комунікації. Основна ідея технології MMX полягає в одночасному обробленні декількох елементів даних за одну інструкцію – так звана технологія SIMD (*Single Instruction* –

Multiple Data). Розширення MMX використовує нові типи цілочисельних даних, які можна записати в 64-бітних регістрах: вісім байтів, чотири слова, два подвійних слова або одне 64-бітне слово. Такі формати даних обробляються в 64-бітних регістрах MMX-розширення. Регістри MMX0 ... MMX7 представляють собою молодші 64 біта 80-бітних регістрів FPU. Блок MMX є присутнім практично на всіх сучасних процесорах.

Кожна інструкція MMX виконує дію відразу над усім комплектом операндів, розміщених у регістрах. Ще одна особливість технології MMX – підтримка арифметики з насиченням (*saturating arithmetic*). Її застосовують для оброблення аудіосигналів, тому що, на відміну від звичайної арифметики із циклічним переповненням (*wraparound mode arithmetic*), арифметика з насиченням не так сильно змінює звучання аудіосигналу. Ті ж переваги мають місце й при обробленні кольорів зображення.

Блок MMX (і підтримка відповідних інструкцій) є на всіх сучасних процесорах, починаючи з 6-го покоління. У розширенні SSE набір цілочисельних інструкцій для блока MMX розширений.

Розширення SSE (Streaming SIMD Extensions – потокові SIMD-розширення) призначене для прискорення оброблення великих потоків даних у форматі із рухомою крапкою. Потокове розширення реалізується на апаратному блоці XMM, у якому спочатку було вісім 128-бітних регістрів XMM0 ... XMM7. У 64-бітних розширеннях число регістрів XMM збільшили до 16. У процесорі Pentium 4 набір інструкцій одержав чергові розширення – SSE2, а пізніше й SSE3, що призвело до додавання нових типів 128-бітних операндів для блока XMM. Використання блока XMM дозволяє суміщати виконання в режимі SIMD інструкцій із цілочисельними даними і даними із рухомою крапкою.

Блок дозволяє виконувати векторні (вони ж пакетні) і скалярні інструкції. Векторні інструкції реалізують операції відразу над усіма комплектами операндів. Скалярні інструкції працюють із одним комплектом операндів – молодшим словом. Блок XMM і розширення SSE, SSE2 і SSE3 використовуються й у сучасних процесорах AMD.

5.3. ОРГАНІЗАЦІЯ ПАМ'ЯТІ

Фізична пам'ять, до якої процесор має доступ по шині адреси, називається оперативною пам'яттю (чи оперативним запам'ятовуючим пристроєм – ОЗП). На найнижчому рівні пам'ять комп'ютера можна розглядати як масив бітів. Один біт може зберігати значення 0 або 1. Але процесору незручно працювати з пам'яттю на рівні бітів, тому ОЗП організовано як послідовність комірок довжиною в байт. Один байт складається з 8-ми бітів. Кожному байту ОЗП відповідає своя унікальна

адреса (його номер), яку називають фізичною адресою комірки. Діапазон значень фізичних адрес залежить від розрядності шини адреси процесора. Для i486 і Pentium він знаходиться в межах від 0 до $2^{32}-1$ (4 Гб). Для процесорів Pentium Pro/II/III/IV цей діапазон ширший – від 0 до $2^{36}-1$ (64 Гб). Механізм управління пам'яттю повністю апаратний. Це означає, що програма не може сама сформувати фізичну адресу пам'яті на адресній шині. Їй доводиться «грати» за правилами процесора.

Процесор апаратно підтримує дві моделі використання оперативної пам'яті: сегментну модель, при використанні якої в програмі виділяються безперервні області пам'яті (сегменти), а сама програма може звертатися тільки до даних, які знаходяться в цих сегментах; сторінкову модель, яку можна розглядати як надбудову над сегментною моделлю. При сторінковій моделі оперативна пам'ять розглядається як сукупність блоків фіксованого розміру (4 Кб і більше). Основне застосування цієї моделі пов'язане з організацією віртуальної пам'яті, що дозволяє операційній системі залучати для роботи програм простір пам'яті набагато більше, ніж об'єм фізичної пам'яті.

Такий підхід до управління пам'яттю забезпечує:

- компактність зберігання адреси в машинній команді;
- гнучкість механізму адресації;
- захист адресних просторів завдань у багатозадачній системі;
- підтримку віртуальної пам'яті.

Сегментна модель пам'яті

Сегментація – це механізм адресації, що забезпечує існування декількох незалежних адресних просторів як у межах одного завдання, так і в системі в цілому для захисту завдань від взаємного впливу. В основі механізму сегментації, який підтримується на апаратному рівні, лежить поняття сегмента.

Сегментна адресація – це схематична логічна адресація пам'яті в архітектурі процесора. Лінійна адреса визначеної комірки пам'яті, яка в певних режимах буде співпадати з фізичною адресою, складається із двох частин: сегмента та зміщення.

Сегментом називається умовно виділена область адресного простору визначеного розміру, а зміщенням – адреса комірки пам'яті відносно початку сегмента. Базою сегмента називається лінійна адреса (адреса відносно всього об'єму пам'яті), яка вказує на початок сегмента в адресному просторі. Результатом буде сегментна (логічна) адреса, яка відповідає лінійній адресі (*база сегмента + зміщення*) і виставляється процесором на шину адреси. *Селектором* називається число (у x86 – 16-бітне), яке однозначно визначає сегмент. Селектор записується в сегментні регістри.

У реальному і захищеному режимі роботи процесора функціонування сегментної адресації різне.

Для процесорів Intel, починаючи з i8086, прийнятий особливий підхід до управління пам'яттю. Кожна програма в загальному випадку може складатися з будь-якої кількості сегментів, але безпосередній доступ вона має тільки до трьох основних сегментів (*сегмента коду, сегмента даних і сегмента стека*), а також до *додаткових сегментів даних* числом від одного до трьох.

Програма заздалегідь не знає, за якими фізичних адресах будуть розміщені її сегменти. Цим займається операційна система. Операційна система розміщує сегменти програми в оперативній пам'яті за певними фізичними адресами, після чого поміщає значення цих адрес у певні місця. Куди саме, залежить від режиму роботи процесора. Так, у реальному режимі ці адреси заносяться безпосередньо у відповідні сегментні реєстри, а в захищеному режимі вони потрапляють в елементи спеціальної *системної таблиці дескрипторів*. У середині сегмента програма звертається до адрес відносно початку сегмента лінійно, тобто починаючи з 0 і закінчуючи адресою, яка дорівнює розміру сегмента. Ця відносна адреса (зміщення), яку процесор використовує для доступу до даних усередині сегмента, називається *ефективною*.

Розрізняють три основні моделі сегментної організації пам'яті в різних режимах:

- сегментна модель пам'яті реального режиму;
- сегментна модель пам'яті захищеного режиму;
- безперервна модель пам'яті захищеного режиму.

Відмінності моделей добре простежуються на схемах (див. рис. 5.5–5.7). Розглянемо порядок формування фізичної адреси в реальному і захищеному режимах. Під фізичною адресою розуміється адреса пам'яті, що видається на шину адреси процесора. З ним пов'язана назва іншої адреси, з якою працює процесор – лінійна адреса. Ці назви є синонімами тільки при відключенні сторінкового перетворення адрес. У реальному режимі сторінкова адресація завжди відключена, тому значення лінійної і фізичної адрес є однаковими.

У захищеному режимі сторінкова адресація, як правило, включена, відповідно і значення лінійної і фізичної адрес є різними числами, які формуються за певними правилами. Ще раз акцентуємо, що сторінкова модель є надбудовою над сегментною моделлю.

Далі розглянемо схеми формування фізичних адрес у реальному і захищеному режимах роботи процесора. На схемі (див. рис. 5.5) відображено механізм сторінкового перетворення адрес у реальному режимі. Цей механізм поєднує дві принципово різні моделі організації оперативної пам'яті і видає на

шину адреси безпосереднє значення фізичної адреси. У схемі для захищеного режиму цей механізм не відображено, але його присутність мається на увазі.

Формування фізичної адреси в реальному режимі (див. рис. 5.5, 5.8)

Перелічимо характеристики механізму адресації фізичної пам'яті в реальному режимі:

- діапазон зміни фізичної адреси – від 0 до 1 Мб. Ця величина визначається тим, що шина адреси i8086 мала 20 ліній;

- максимальний розмір сегмента – 64 Кб. Причина обмеження полягає в 16-розрядній архітектурі i8086. Неважко підрахувати, що максимальне значення, яке можуть прийняти 16-розрядні регістри, складає $2^{16}-1$, що стосовно пам'яті і визначає величину 64 Кб.

Для звернення до конкретної фізичної адреси оперативної пам'яті необхідно визначити адресу початку сегмента (сегментну складову) і зміщення усередині сегмента.

Поняття адреси початку сегмента є принципово важливим і вимагає додаткового пояснення. Виходячи з розрядності сегментних регістрів можна стверджувати, що сегментна складова адреси (чи база сегмента) є всього лише 16-розрядним значенням, поміщеним в один із сегментних регістрів. Максимальне значення, яке при цьому виходить, дорівнює $2^{16}-1$. Виходить, що адреса початку сегмента може бути тільки в діапазоні 0 ... 64 Кб від початку оперативної пам'яті. Виникає питання, як адресувати іншу частину оперативної пам'яті аж до 1 Мб з урахуванням того, що розмір самого сегмента не перевищує 64 Кб. Річ у тому, що в сегментному регістрі містяться тільки старші 16 біт фізичної адреси початку сегмента. Не вистачає молодших чотирьох бітів 20-розрядної адреси, які виникають як результат зсуву значення в сегментному регістрі вліво на 4 розряди. Ця операція зсуву виконується апаратно і для програмного забезпечення абсолютно прозора.

20-розрядне значення, яке ми отримали в результаті, і є справжньою фізичною адресою, яка відповідає початку сегмента. Що стосується другого компонента (зміщення), що бере участь в утворенні фізичної адреси деякого об'єкта в пам'яті, то він є 16-розрядним числом. Це значення може міститися явно в команді або побічно в одному з регістрів загального призначення. Процесор самостійно складає ці дві складові, у результаті виходить фізична адреса пам'яті завдовжки 20 біт. Цей механізм утворення фізичної адреси дозволяє зробити програмне забезпечення універсальним, тобто незалежним від конкретних адрес завантаження його в оперативній пам'яті (див. рис. 5.8).

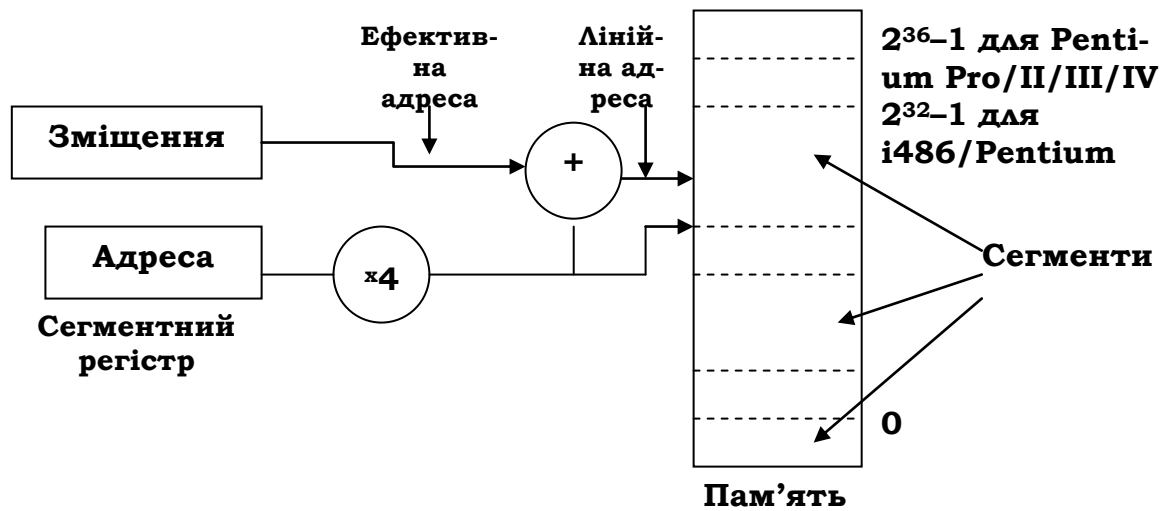


Рис. 5.5. Сегментна модель пам'яті в реальному режимі роботи процесора

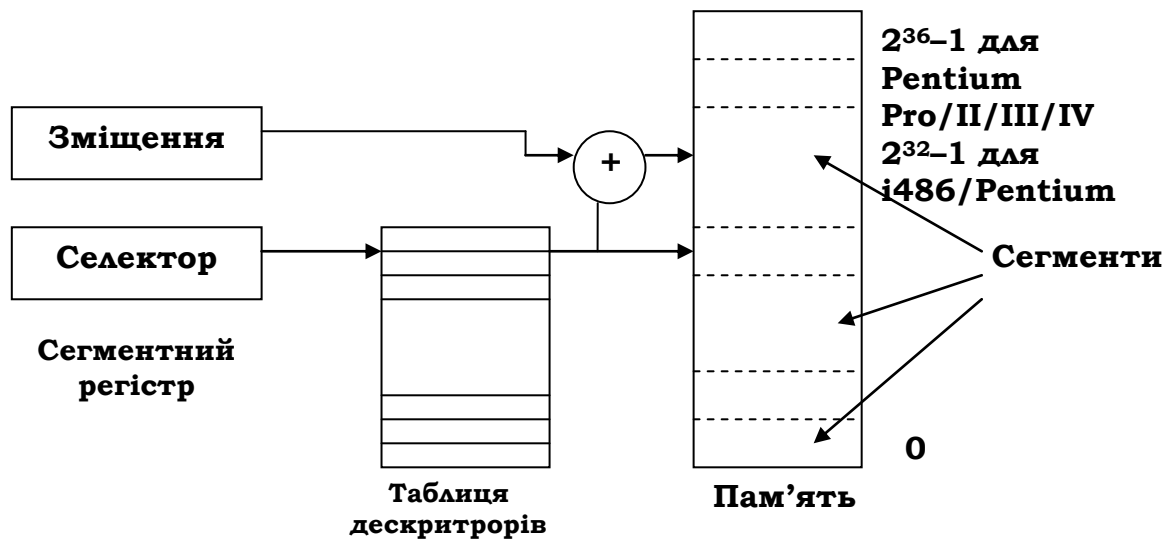


Рис. 5.6. Сегментна модель пам'яті захищеного режиму

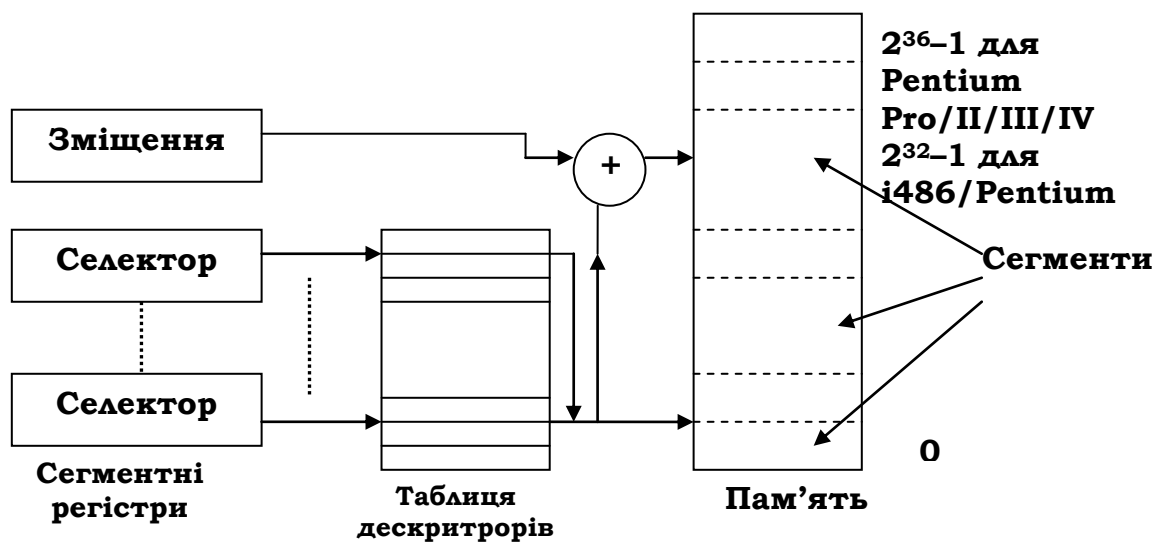


Рис. 5.7. Безперервна модель пам'яті захищеного режиму

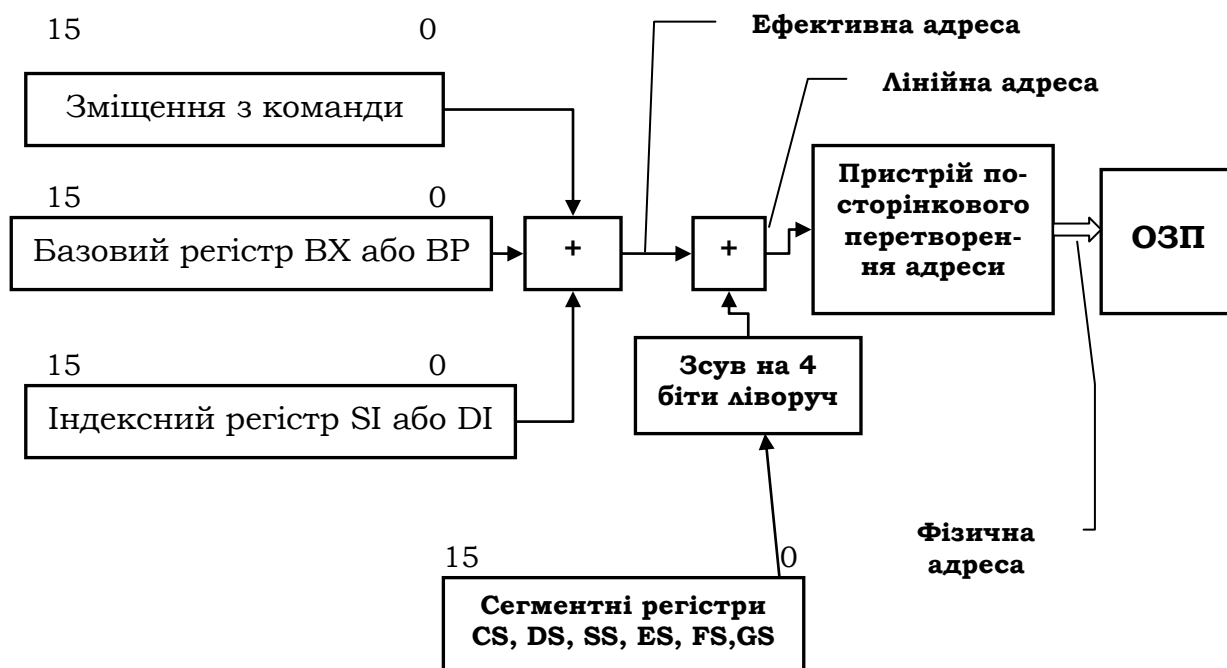


Рис. 5.8. Механізм формування фізичної адреси в реальному режимі

На рис. 5.8 відображено формування деякої цільової фізичної адреси: сегментна частина надходить з одного із сегментних реєстрів, над нею здійснюється зсув на чотири розряди вліво і після цього вона сумується із зміщенням. У свою чергу, видно, що значення зміщення можна отримати мінімум з одного і максимум із трьох джерел: зі значення зміщення в самій машинній команді; і/або із вмісту одного базового; і/або із вмісту одного із індексних реєстрів. Кількість джерел, що беруть участь у формуванні зміщення, визначається кодуванням конкретної машинної команди, і якщо таких джерел декілька, то значення в них складаються. Слід зауважити, що не варто хвилюватися через невідповідність розмірів шини адреси процесора i486 або Pentium (32 біта) і 20-розрядного значення фізичної адреси реального режиму. Поки процесор знаходиться в реальному режимі, старші 12 ліній шини адреси просто недоступні, хоча за певних умов і існує можливість роботи з першими 64 Кб оперативної пам'яті, які розташовані після першого мегабайта.

Такий спосіб організації пам'яті налічує ряд недоліків:

- 1) сегменти безконтрольно розміщуються з будь-якої адреси, кратної 16-ти (оскільки вміст сегментного реєстра апаратно зсувається на 4 розряди), і, як наслідок, програма може звертатися за будь-якими адресами, у тому числі й реально не існуючими;
- 2) сегменти мають максимальний розмір 64 Кб;
- 3) одні сегменти можуть перекриватися з іншими сегментами.

Бажаннтям ввести в архітектуру засоби, що дозволяють позбавитися від вказаних недоліків, і обумовлена поява

захищеного режиму, у якому працюють усі сучасні операційні системи, у тому числі Windows і Linux.

Формування фізичної адреси в захищеному режимі

Основна ідея захищеного режиму – захистити програми, які виконує процесор, від взаємного впливу. У захищеному режимі процесор реалізує різні механізми безпеки (див. рис. 5.5, 5.9). Розглянемо механізм захисту доступу до пам'яті. Для введення будь-якого охоронного механізму треба мати якомога більше інформації про об'єкти, що охороняються. Для процесора такими об'єктами є виконуваний ним програми. Організуючи захист програм за доступом до пам'яті, фірма Intel не стала порушувати принцип сегментації, властивий її процесорам. Оскільки кожна програма займає один або кілька сегментів у пам'яті, то логічно мати більше інформації про ці сегменти як про об'єкти, реально існуючі в даний момент в обчислювальній системі. Якщо кожному із сегментів присвоїти певні атрибути, то частина функцій з контролю за доступом до них можна перекласти на процесор. Що й було зроблено. Будь-який сегмент пам'яті в захищеному режимі має наступні основні атрибути:

- місце розташування сегмента в пам'яті;
- розмір;
- рівень привілеїв (визначає права цього сегмента відносно інших сегментів);
- тип доступу (визначає призначення сегмента).

На відміну від реального режиму, у захищеному режимі програма вже не може звернутися за будь-якою лінійною адресою пам'яті. Для цього вона повинна мати певні повноваження і відповідати ряду вимог. Ключовим об'єктом захищеного режиму є спеціальна структура – *дескриптор сегмента*, який є 8-байтовим дескриптором (короткий опис) безперервної області пам'яті, яка має перераховані вище атрибути (див. рис. 5.9).

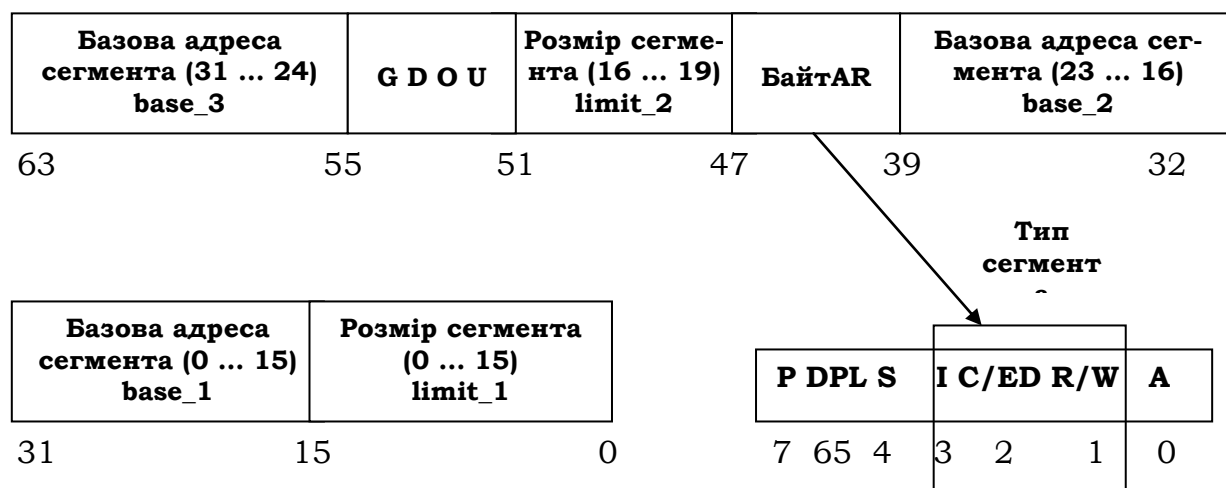


Рис. 5.9. Структура дескриптора сегмента захищеного режиму процесора

Розглянемо призначення деяких полів дескриптора сегмента:

1) `limit_1` і `limit_2` – разом організують 20-розрядне поле, що визначає розмір сегмента. Поле `limit_1` зберігає розряди від 0 до 15, поле `limit_2` зберігає розряди від 16 до 19;

2) `base_1`, `base_2`, `base_3` – разом організують 32-розрядне поле, що містить значення лінійної адреси початку сегмента в пам'яті;

3) `AR` – байт, біти якого визначають права доступу до сегмента;

4) `D` – біт розрядності операндів і адрес;

5) `G` – біт гранулярності.

У захищеному режимі розмір сегмента не фіксований, його розташування можна задати в межах 4 Гб. Якщо подивитися на рис. 5.9, то виникне питання: чому розірвані поля, що визначають розмір сегмента і його початкову (базову) адресу? Це результат еволюції процесорів. Захищений режим вперше з'явився в процесорі i80286. Цей процесор мав 24-розрядну адресну шину і, відповідно, міг адресувати в захищеному режимі до 16 Мб оперативної пам'яті. Для цього йому досить було мати в дескрипторі поле базової адреси в 24 біта і поле розміру сегмента завдовжки 16 бітів. Після появи процесора i80386 з 32-розрядними шинами команд і даних з метою сумісності програм розробники не стали змінювати формат дескриптора, а просто використали вільні поля. Усередині процесора ці поля об'єднані. Зовні ж вони залишилися розділеними, з чим доводиться погоджуватись при програмуванні.

Наступний цікавий момент пов'язаний із тим, що розмір сегмента в захищеному режимі може досягати 4 Гб, тобто сегмент здатний займати весь доступний фізичний простір пам'яті. Як це можливо, якщо сумарний розмір поля розміру сегмента складає всього 20 бітів, що відповідає величині 1 Мб? Секрет прихований у полі гранулярності – біт `G` (див. рис. 5.9). Якщо `G` = 0, то значення в полі розміру сегмента є розміром сегмента в байтах, якщо `G` = 1 – у сторінках. Розмір сторінки дорівнює 4 Кб. Неважко підрахувати, що при максимальному значенні поля розміру сегмента `0ffffh` це відповідає 1 Мбайт сторінок або величині $1 \text{ Мб} \times 4 \text{ Кб} = 4 \text{ Гб}$.

Виведення інформації про базову адресу сегмента і його розмір на рівень процесора дозволяє апаратно контролювати, як програми працюють з пам'яттю і запобігати зверненням до адрес, що знаходяться поза сегментом. Інший аспект захисту полягає в тому, що сегменти нерівноправні в правах доступу до них. Інформація про це міститься в спеціальному байті `AR`, що входить до складу дескриптора. Тип сегмента визначають наступні елементи байта `AR`: поле `DPL`, біти `R/W`, `C/ED` і `I`. Поле `DPL` – частина механізму захисту за привілеями. Цей механізм полягає в тому, що конкретний сегмент може знаходитися на

одному з чотирьох рівнів привілейованості (тобто мати номер 0, 1, 2 або 3). Максимум можливостей надається рівню 0. Існує ряд обмежень (знову-таки на апаратному рівні) на взаємодію сегментів коду, даних і стека з різними рівнями привілеїв.

Таким чином, у захищеному режимі перед використанням будь-якої області пам'яті має бути проведена певна робота за ініціалізації відповідного дескриптора. Цю роботу виконує операційна система або програма, сегменти якої також описуються подібними дескрипторами. Усі дескриптори збираються разом в одну з трьох таблиць:

1) глобальна таблиця дескрипторів (*Global Descriptor Table – GDT*), адреса якої зберігається в регістрі GDTR;

2) локальна таблиця дескрипторів (*Local Descriptor Table – LDT*), її адреса знаходиться в регістрі LDTR;

3) таблиця дескрипторів векторів переривань (*Interrupt Descriptor Table – IDT*), її адреса зберігається в регістрі IDTR.

У яку саме таблицю має бути поміщений дескриптор, визначається його призначенням. Адреса, за якою розміщуються ці дескрипторні таблиці, може бути будь-яка; вона зберігається в спеціально призначеному для цього системному регістрі. Схеми, показані на рис. 5.5–5.8, ілюструють принцип формування адреси в захищеному режимі. Важливо відмітити зміну ролі сегментних регістрів. У захищеному режимі вони містять не адресу, а селектор, тобто покажчик на відповідну комірку однієї з таблиць дескрипторів (GDT або LDT).

5.4. ФУНКЦІОНАЛЬНА КЛАСИФІКАЦІЯ МАШИННИХ КОМАНД

Набір інструкцій (система команд) сучасних процесорів x86 вбирає в себе інструкції всіх попередніх поколінь. Інструкції можна розділити на прикладні, які використовуються для програмування прикладних задач, і системні, які використовуються ОС для створення середовища, у якому працюють прикладні задачі.

Прикладні інструкції можна розділити на декілька груп.

1. Інструкції загального призначення – основні цілочисельні інструкції x86, які використовуються практично всіма програмами. Ці інструкції завантажують, зберігають і обробляють дані, розташовані в регістрах загального призначення і пам'яті. Частина інструкцій служить для зміни послідовності виконання алгоритму – це інструкції умовних і безумовних переходів, виклики процедур. Базові інструкції загального призначення реалізовані у всіх процесорах x86 (за винятком деяких інструкцій, наявність яких визначається через інструкцію CPUID). Інструкції 64-розрядних режимів (*long mode instructions*) з'явилися тільки в процесорах із 64-розрядними розширеннями і доступні лише у відповідних режимах.

2. Інструкції співпроцесора x87 для чисел із рухомою точкою працюють зі співпроцесором (FPU – *Floating Point Unit*), вони використовуються в застарілих задачах для програмування точних обчислень. Відмітимо, що ці інструкції підтримують різні формати даних із рухомою крапкою (FP-чисел): 80 бітів – розширена точність, 64 біти – подвійна точність, 32 біти – одинарна точність. Крім того, підтримуються двійково-десяткові формати і перетворення форматів. 80-бітовий FP-формат забезпечує максимальну точність і максимальний діапазон чисел. Набір інструкцій x87 поступово розширювався, у процесорі P6 з'явилися інструкції умовних переходів, що скорочують число розгалужень у програмі.

Медіа-інструкції 64-розрядних розширень оперують із даними, розташованими в 64-розрядних регістрах MMX. Вони виконують цілочисельні операції і обчислення з рухомою крапкою в скалярному і векторному варіантах і призначені для медіа-застосувань, що працюють із блоками даних. Векторні цілочисельні інструкції з'явилися в початковому наборі MMX, їх склад розширювався інструкціями AMD Extension to MMX і частиною інструкцій SSE (введених Intel). Векторні інструкції з рухомою крапкою з'явилися в 3DNow!, їх набір розширився в AMD Extensions to 3DNow! Instructions.

Медіа-інструкції 128-розрядних розширень оперують даними, розташованими в 128-розрядних регістрах XMM. Вони виконують цілочисельні операції і обчислення з рухомою крапкою в скалярному і векторному варіантах. Ці інструкції призначені для високопродуктивних медіа- і наукових задач, що працюють із блоками даних. У потоковому розширенні SSE визначені інструкції з рухомою крапкою одинарної точності (32 біти), у SSE2 введені інструкції подвійної точності (64 біти), у SSE3 введено 13 додаткових інструкцій (до тих, що існують у SSE і SSE2), включаючи SIMD-інструкції з FP-числами подвійної точності, цілими числами, а також інструкції управління пам'яттю і кешуванням.

Група інструкцій забезпечує зв'язки між вищенаведеними групами. Вони забезпечують пересилку даних між блоками MMX, XMM і загальними регістрами. Ряд інструкцій (як цілочисельних, так і з рухомою крапкою) можна виконувати як у блоках MMX (і FPU), так і в блоках XMM.

Події (переривання і виключення) в архітектурі IA-32

Переривання (*interrupt*) і виключення (*exception*), які узагальнено називають подіями (*event*), порушують нормальний хід виконання програми для оброблення зовнішніх подій або сигналів про виникнення особливих умов або помилок. Після

виникнення події процесор зберігає в стеку регістр EFLAGS/FLAGS (див. рис. 5.1) і покажчик CS:EIP/IP на ту інструкцію, яку він повинен буде виконати після оброблення події. Цією інструкцією буде наступна за тією, під час виконання якої відбулася подія, або та ж сама. У захищеному режимі при виникненні ряду виключень у стеку зберігається ще й код помилки. Після збереження цих значень процесор переходить до виконання програми – обробника даної події, визначаючи точку входу в нього через номер (0 ... 255) за таблицею переривань. Номер елементу в таблиці переривань називається **вектором переривання**, він визначається джерелом події. Обробник події повинен закінчуватися спеціальною інструкцією повернення IRET, за якою зі стека відновлюються покажчик CS:EIP/IP і колишні значення прапорів. Для виключень, у яких зберігається і код помилки, обробник до виконання інструкції IRET повинен витягнути зі стека код помилки.

Розрізняють три типи подій:

1) виключення (внутрішні переривання) процесора і співпроцесора, вектор стану програми (див. розд. 2.8) визначається типом події, що відбулася;

2) немасковані зовнішні переривання по входу NMI, вектор стану фіксований (01);

3) масковані зовнішні переривання по входу INT, вектор стану передається контролером переривань.

Реакція процесора на переривання може бути заборонена (точніше, відкладена) скиданням прапора IF (процесор відреагує на переривання, коли прапор IF буде встановлений).

Для переривань, що викликаються програмно, вектор визначається в команді.

Інші події з цього списку в прямому розумінні перериваннями не є, оскільки є лише специфічним механізмом виклику процедур – не за адресою, а за номером у таблиці переривань, причому незалежно від стану прапора IF. Програмні переривання широко використовуються для викликів сервісів BIOS і ОС – це компактно і зручно. Наприклад, програмне переривання INT 3 (однобайтна інструкція, на відміну від інших – двобайтових) застосовується для розстановки точок HLT у відладчиках (вбудовані засоби відладки виробляють виключення з вектором 01).

Виключення (внутрішні переривання) процесора генеруються при виникненні особливих умов виконання поточної інструкції. У більшості своїй вони неочікувані. Виключення обробляються процесором залежно від їх походження.

Виключення підрозділяються на три типи:

1) відмова (*fault*) – виключення, що виникає до виконання інструкції, коли процесор виявляє неможливість її виконання. До відмов відноситься, наприклад, спроба звернення до відсутньої сторінки пам'яті (використовується для організації віртуальної пам'яті);

2) пастка (*trap*) – виключення, що виникло в результаті виконання інструкції (наприклад, ділення на нуль);

3) аварія (*abort*) – виключення, для якого неможливо точно визначити інструкцію, що спричинила його (і неможливі коректні дії для продовження роботи). До цих виключень відносяться подвійна відмова (виключення при обробленні виключення) і виключення від засобів машинного контролю (*machine-check*).

Масковані зовнішні переривання обробляються процесором за сигналом на вході INT тільки при встановленому прапорі дозволу переривань IF.

Немасковані переривання обробляються процесором незалежно від стану прапора дозволу переривання IF. До них відносяться переривання, що приходять по лінії NMI (*Non-Maskable Interrupt*), а для процесорів, що підтримують режим системного управління, ще й по лінії SMI (*System Management Interrupt*).

Кожному номеру (0 ... 255) переривання або виключення відповідає елемент у таблиці дескрипторів переривань (*Interrupt Descriptor Table* – IDT), який трактується залежно від режиму процесора.

У реальному режимі таблиця IDT містить дальні адреси (подвійні слова) обслуговуючих процедур. У захищеному режимі таблиця IDT містить 8-байтові дескриптори переривань, може зберігати від 32 до 256 дескрипторів і розташовуватися в будь-якому місці фізичної пам'яті. Дескриптори можуть бути шлюзами, або вентилями (*gates*), переривань, пасток або задач. Шлюзи переривань і пасток служать для виклику процедур оброблення, розташованих у сегментах, описаних їх дескрипторами. Ці шлюзи забезпечують захист при переданні управління: процесор передає управління обробникові тільки при певних співвідношеннях привілеїв програм, що перериваються, програм-обробників і привілею дескриптора. Програмний код обробника переривань повинен бути не менш привілейованим, ніж код задачі, що переривається (інакше спрацює виключення захисту). Якщо перериваюча процедура виконується на більш високому рівні привілеїв, то процесор автоматично перемикає стек (це додаткова витрата часу на звернення до пам'яті). По перериванню (у тому числі й апаратному) можливе і перемикання задач, для цього дескриптор переривання повинен бути шлюзом завдання.

У 64-бітовому режимі (*long mode*) дескриптори мають нову 16-байтову структуру, і тут інші правила перемикування стека. Крім того, у цьому режимі не виконується апаратне перемикування завдань.

На початку обробленні будь-якого (у тому числі й програмного) переривання процесор автоматично скидає прапор дозволу переривань IF. Процедура оброблення завершується інструкцією IRET, за якою зі стека відновлюються автоматично збережені регістри (у відновленому регістрі прапорів переривання дозволени), і процесор починає виконання інструкції, наступної за тією, після якої виконувалося переривання. Звичайно, програмно під час обслуговування переривання можлива умисна або випадкова зміна покажчика або вмісту стека, і тоді інструкція IRET «відправляє» процесор за іншою адресою, внаслідок чого комп'ютер може зависнути. Якщо на час оброблення потрібна реакція і на інші переривання, обробник повинен встановити прапор IF. Переривання, що обслуговуються до завершення оброблення попереднього, називаються вкладеними. Вкладені переривання небезпечні через можливість переповнювання стека, оскільки кожне «вкладення» задіюватиме стек для своїх цілей. Переповнювання стека може також бути причиною зависань. Довгі процедури оброблення із скинутим прапором IF можуть привести до втрати системного часу, оскільки «годинник» операційної системи використовують апаратні переривання від таймера. Процедура обслуговування для кожного джерела апаратних переривань повинна бути написана дуже обачно і враховувати нюанси роботи решти підсистем. Загальне правило – процедура повинна виконувати мінімальні дії, а якщо потрібна програмно складна реакція, то процедура повинна надіслати повідомлення про подію, яка оброблятиметься пізніше (не обробником переривання).

Для оброблення апаратних переривань у багатопроцесорних системах традиційні апаратні засоби стають неприйнятними. Для вирішення цієї проблеми в процесори, починаючи з другого покоління Pentium, введений удосконалений програмований контролер переривань (*Advanced Programmable Interrupt Controller* – APIC). Локальні переривання обслуговуються тільки тим процесором, на який надходять їх запити. Загальні (які можуть розділятися) переривання надходять до процесорів у вигляді повідомлень по інтерфейсу APIC. При цьому контролери заздалегідь програмуються, чим визначаються функції кожного з процесорів у разі виникнення того або іншого апаратного переривання. Контролери APIC кожного з процесорів і контролер системної плати, зв'язані інтерфейсом APIC, виконують

маршрутизацію переривань (*interrupt routing*), причому як статичну, так і динамічну. Зовні програмний інтерфейс оброблення переривань залишається сумісним із попередніми контролерами, що забезпечує прозорість присутності АРІС для прикладного програмного забезпечення.

5.5. СИСТЕМА КОМАНД ПРОЦЕСОРА

Сучасні асемблери значно спрощують процес написання програм та приховують наявний формат машинних команд процесора. Асемблер можна вважати сьогодні мовою високого рівня, хоча й найбільш низькорівневою серед інших. Програмуванням безпосередньо в машинних кодах можна створювати програми, які не дають можливості втручання у свою роботу або «дизасемблювання».

Відповідно і розібратись в алгоритмах, записаних мовою кодових комбінацій, дуже важко. Знань лише кодів команд, які даються в технічній документації розробниками, може виявитись недостатньо. Документація, як правило, не розкриває тонкощів інтерпретації команд мікропроцесором та особливостей їх виконання. Але вміння оперувати машинними командами є основним хакерським інструментом і, звичайно, необхідне для побудови систем програмного захисту або відладки програм.

Наприклад, система команд Pentium IV має близько 330 мнемонік машинних команд. Якщо ж рахувати по кодах операцій, то їх буде значно більше. Увесь набір машинних команд можна розбити на наступні групи:

- 1) команди загального призначення, що виконуються цілочисельним пристроєм, у тому числі системні команди;
- 2) команди співпроцесора;
- 3) команди MMX-розширення;
- 4) команди XMM-розширення (набори команд SSE і SSE2).

У межах кожної з цих великих груп виходячи з функціонального призначення окремих команд можна провести подальше розбиття на підгрупи.

Команди загального призначення

Командами цієї категорії охоплюється базовий набір операцій з оброблення даних. Це переміщення даних, виконання арифметичних і логічних дій над даними і т. ін. Для виконання команд цієї групи використовуються регістри загального призначення, сегментні регістри, регістр EFLAGS. Для того щоб уявити можливості даної групи команд, нижче наведена класифікація цих команд з розбиттям на підкатегорії:

- команди пересилки даних;
- команди двійкової арифметики;

- команди десятикової арифметики;
- логічні команди;
- команди зсуву і інверсії;
- бітові команди;
- команди управління програмним потоком;
- ланцюжкові команди;
- команди для роботи з регістром прапорів;
- команди для роботи з сегментними регістрами;
- системні команди і відособлені команди (поза групами).

Для уточнення змісту машинних команд розглянемо приклади команд асемблера.

Команди пересилки даних і адрес

Пересилка даних без перетворення:

MOV	Пересилка операнда
CMOVcc	Умовна пересилка операнда
SETcc	Умовна установка байта
PUSH	Запис операнда в стек
PUSHA(D)	Запис у стек вмісту всіх регістрів
POP	Читання операнда зі стека
POPA(D)	Читання зі стека вмісту всіх регістрів
XCHG	Обмін між регістрами або пам'яттю і регістром
XLAT	Перетворення кодів
BSWAP	Перестановка байтів

Пересилка даних із перетворенням:

MOVSX	Пересилка байта або слова з розширенням знаку
MOVZX	Пересилка байта або слова з розширенням нулями

Введення/виведення даних:

IN	Введення операнда з порту в акумулятор
OUT	Виведення операнда з акумулятора в порт

Завантаження ефективної адреси і селекторів:

LEA	Завантаження ефективної адреси ЕА в регістр
LDS	Завантаження покажчика адреси для сегмента DS
LES	Завантаження покажчика адреси для сегмента ES
LFS	Завантаження покажчика адреси для сегмента FS
LGS	Завантаження покажчика адреси для сегмента GS
LSS	Завантаження покажчика адреси для сегмента SS

Арифметичні операції

Складання:

ADD	Складання операндів
ADC	Складання операндів з ознакою CF (прапор перенесення)
XADD	Обмін операндами і складання
INC	Інкремент операнда
AAA	ASCII-корекція результату складання
DAA	Десяткова корекція результату складання

Віднімання:

SUB	Віднімання операндів
SBB	Віднімання операндів і ознаки CF (позики)
DEC	Декремент операнда
NEG	Зміна знаку операнда (з переведенням у додатковий код)
AAS	ASCII-корекція результату віднімання
DAS	Десяткова корекція результату віднімання

Порівняння:

CMP	Порівняння операндів
CMPSXCHG	Порівняння і обмін операндами
CMPSXCHG8B	Порівняння і обмін збільшеними учетверо словами (8 байтів)

Множення:

MUL	Беззнакове множення
IMUL	Знакове (цілочисельне) множення
AAM	ASCII-корекція результату множення

Ділення:

DIV	Беззнакове ділення
IDIV	Знакове (цілочисельне) ділення
AAD	ASCII-корекція результату ділення

Зміна розрядності шляхом розширення знаку:

CBW	Перетворення байта (AL) в слово (AX)
CWDE	Перетворення слова (AX) в подвійне слово (EAX)
CWD	Перетворення слова (AX) в подвійне слово (DX, AX)
CDQ	Перетворення подвійного слова (EAX) в збільшене учетверо слово (EDX, EAX)

Логічні операції

NOT	Інверсія операнда (логічне «ні»)
AND	Кон'юнкція операндів (логічне «і»)
OR	Диз'юнкція операндів (логічне «або»)
XOR	Нерівнозначність операндів (виключне «або»)
TEST	Логічне порівняння операндів (установка ознак ZF, SF, PF)

Зсуви

SHL/SAL	Зсув вліво
SHR	Логічний зсув вправо
SAR	Арифметичний зсув вправо
SHLD	Двооперандний зсув вліво
SHRD	Двооперандний зсув вправо
ROL	Циклічний зсув вліво
ROR	Циклічний зсув вправо

RCL	Циклічний зсув вліво через прапор перенесення (CF)
RCR	Циклічний зсув вліво через прапор перенесення (CF)

Операції з бітами

BT	Перевірка біта
BTS	Перевірка і установка біта
BTR	Перевірка і скидання біта
BTC	Перевірка та інверсія біта
BSF	Пряме сканування бітів
BSR	Зворотне сканування бітів
Scc	Умовна установка байтів

Операції з рядками символів

LODS	Завантаження символу в акумулятор
STOS	Запис символу з акумулятора
INS	Введення символу
OUTS	Виведення символу
MOVS	Пересилка символу
CMPS	Порівняння символу
SCAS	Сканування рядка символу

5.6. СТРУКТУРА МАШИННОЇ КОМАНДИ МІКРОПРОЦЕСОРІВ

INTEL 80X86

Машинна команда є закодованою за певними правилами вказівкою мікропроцесору на виконання деякої операції або дії. Наведений на рис. 5.6 формат машинної команди є найзагальнішим. Максимальна довжина машинної команди – 15 байтів. Реальна команда може містити різну кількість полів, аж до одного – тільки Код операції (КОП).

Поле *Префікс* може містити до 4-х префіксів (може не містити жодного). Кожний префікс займає 1 байт. Поле *КОП* містить одно- або 2-байтний код інструкції. Поле *mod r/m* займає один байт, задає спосіб адресації та специфікацію регістрів, що використовуються. Поле *SIB* (Scale-Index-Base – масштаб-індекс-база) уточнює спосіб адресації. Залежно від способу адресації поле *Зміщення* займає 1, 2 або 4 байта і зберігає зміщення операнда. Поле *Операнд* зберігає безпосередньо операнд та може складати 1, 2 або 4 байта.

Слід зазначити, що асемблер не розрізняє прописні та строчні букви.

Розглянемо призначення полів машинної команди (див. рис. 5.6) більш детально.

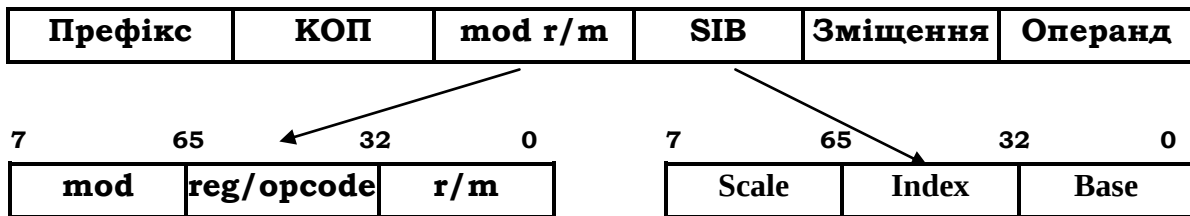


Рис. 5.6. Формат машинної команди

1. Префікси (від лат. *pre* – попереду та *fix* – прикріплений). Необов'язкові елементи машинної команди, кожен з яких складається із одного байта або може бути відсутнім. У пам'яті префікси передують команді. Призначення префіксів – модифікувати операцію, що виконується командою. Прикладна програма може використовувати наступні типи префіксів:

1. *Префікс заміни сегмента*. У явній формі вказує, який сегментний регістр використовується в даній команді для адресації стека або даних. Префікс відмінняє вибір сегментного регістра за умовчанням. Префікси заміни сегмента мають наступні значення:

- 2eh – заміна сегмента cs;
- 36h – заміна сегмента ss;
- 3eh – заміна сегмента ds;
- 26h – заміна сегмента es;
- 64h – заміна сегмента fs;
- 65h – заміна сегмента gs.

2. *Префікс розрядності адреси (67h)* уточнює розрядність адреси (32 або 16-розрядна). Кожній команді, у якій використовується адресний операнд, ставиться у відповідність розрядність адреси цього операнда. Ця адреса може мати розрядність 16 або 32 біти. Якщо розрядність адреси для даної команди 16 біт, це означає, що команда містить 16-розрядне зміщення, яке відповідає 16-розрядному зміщенню адресного операнда відносно початку сегмента. Це зміщення називається ефективною адресою. Якщо розрядність адреси 32 біти, це означає, що команда містить 32-розрядне зміщення. Це відповідає 32-розрядному зміщенню адресного операнда відносно початку сегмента і по його значенню формується 32-бітне зміщення в сегменті. Тобто за допомогою префікса розрядності адреси можна змінити значення розрядності адреси, що діє за умовчанням. Ця зміна стосуватиметься тільки тієї команди, якій передує префікс.

3. *Префікс розрядності операнда (66h)* аналогічний префіксу розрядності адреси, але вказує на розрядність операндів (32 або 16-розрядні), з якими працює команда. Якщо він зустрічається в 32-розрядному полі, то процесор інтерпретує операнди інструкції як 16-розрядні та навпаки.

Значення розрядності адрес і операндів за умовчанням встановлюється за наступним правилом. У реальному режимі і режимі віртуального Intel 80x86 значення цих атрибутів дорівнює 16 бітам. У захищеному режимі значення атрибутів залежать від стану біта D в дескрипторах поточних сегментів (див. рис. 5.5). Якщо $D = 0$, то значення атрибутів, що діють за умовчанням, дорівнює 16 бітам; якщо $D = 1$, то – 32 бітам. Префікс розрядності операнда має значення 66h, а префікс розрядності адреси – 67h. За допомогою префікса розрядності адреси в реальному режимі можна використовувати 32-розрядну адресацію, але при цьому необхідно пам'ятати про обмеженість розміру сегмента значенням 64 Кб. Аналогічно префіксу розрядності адреси можна використовувати префікс розрядності операнда в реальному режимі для роботи з 32-розрядними операндами (наприклад, в арифметичних командах).

4. *Префікс повторення та блокування* використовується з ланцюжковими командами (командами оброблення рядків). Цей префікс «зациклює» команду для оброблення всіх елементів ланцюжка. Система команд підтримує такі типи префіксів:

- *безумовні (f3h)*, які примушують повторюватися ланцюжкову команду декілька разів;

- *умовні (f2h)*, які при зацикленні перевіряють деякі прапори, і за результатами перевірки дозволяють достроковий вихід з циклу;

- *блокування (f0h)*, який вказує процесору про необхідність захвату шини на час виконання машинної команди.

2. Код операції. Обов'язковий елемент, що описує операцію, яка виконується командою. Багатьом командам відповідає декілька кодів операцій, кожен із яких визначає нюанси виконання операції.

Наступні поля машинної команди визначають місцеположення операндів, що беруть участь в операції, і особливості їх використання. Формат цих полів пов'язаний зі способами завдання операндів у машинній команді і буде розглянутий пізніше.

3. Байт режиму адресації *mod r/m*. Значення цього байта визначає форму адреси операндів. Операнди можуть знаходитися в пам'яті в одному або двох регістрах. Якщо операнд знаходиться в пам'яті, то байт *mod r/m* визначає компоненти (зміщення, базовий і індексний регістри), які використовуються для обчислення його ефективної адреси. У захищеному режимі для визначення місцеположення операнда в пам'яті може додатково використовуватися байт SIB (*Scale-Index-Base*). Байт *mod r/m* складається з трьох полів (див. рис. 5.6):

1. поле *mod* визначає кількість байтів, які в команді займає адреса операнда (див. рис. 5.6, поле *Зміщення* в команді). Поле

mod використовується спільно з полем *r/m*, яке вказує на спосіб модифікації адреси операнда (Зміщення в команді). Наприклад, якщо *mod* = 00, це означає, що поле Зміщення в команді відсутнє, і адреса операнда визначається вмістом базового і/або індексного регістра. Які саме регістри використовуватимуться для обчислення ефективної адреси, визначається значенням цього байта. Якщо *mod* = 01, це означає, що поле зміщення в команді присутнє, займає один байт і модифікується вмістом базового і/або індексного регістра. Якщо *mod* = 10, це означає, що поле зміщення в команді присутнє, займає два або чотири байти (залежно від розміру адреси, що діє за умовчанням, або визначається префіксом) і модифікується вмістом базового і/або індексного регістра. Якщо *mod* = 11, це означає, що операндів у пам'яті немає: вони знаходяться в регістрах. Це ж значення байта *mod* використовується у разі, коли в команді застосовується безпосередній операнд;

2. поле *reg/opcode* визначає або регістр, що знаходиться в команді на місці першого операнда, або можливе розширення коду операції;

3. поле *r/m* використовується спільно з полем *mod* і визначає або регістр, що знаходиться в команді на місці першого операнда (якщо *mod* = 11), або базові і індексні регістри, які використовуються для обчислення ефективної адреси (спільно з полем зміщення в команді).

4. Байт SIB (масштаб-індекс-база) використовується для розширення можливостей адресації операндів. Про наявність байта *sib* у машинній команді говорять значення 01 або 10 поля *mod* і значення 100 поля *r/m*. Байт SIB складається з трьох полів, а саме:

1. поля *масштабу ss*. У цьому полі розміщується масштабний множник для індексного компоненту *index*, що займає наступні три біта байта SIB. Поле *ss* може містити одне з наступних значень: 1, 2, 4, 8. При обчисленні ефективної адреси на це значення множиться вміст індексного регістра;

2. поля ***index***, що використовується для зберігання номера індексного регістра, який застосовується для обчислення ефективної адреси операнда;

3. поля ***base***, яке використовується для зберігання номера базового регістра, який також застосовується для обчислення ефективної адреси операнда. Нагадаємо, що як базовий і індексний регістри можуть використовуватися практично всі регістри загального призначення.

5. Поле Зміщення в команді представляє собою 8-, 16- або 32-розрядне ціле число із знаком, що є – повністю або частково

(з урахуванням вищенаведених міркувань) – значенням ефективної адреси операнда.

6. Поле Операнда є необов'язковим. Це поле зберігає 8-, 16- або 32-розрядний безпосередній операнд. Наявність цього поля, звичайно, відбивається на значенні байта *mod r/m*.

У якості операнда в команді може фігурувати константа, яка може бути задана в наступних системах числення:

- двійковій. Послідовність цифр 0 і 1, закінчується буквою *b*. Наприклад, 10111010b;

- вісімковій. Послідовність цифр від 0 до 7, закінчується буквою *q*. Наприклад, 204q;

- десятковій. У звичайній десятковій системі числення константи закінчуються буквою *d*, але не обов'язково. Наприклад, 129d або просто 129;

- шістнадцятковій. Послідовність цифр від 0 до 9 та букв від А до F, закінчується буквою *h*. Першим символом може бути тільки цифра від 0 до 9. Наприклад, 0E23h (у даному випадку перша цифра інформує асемблер про те, що E23 – це число, а не ідентифікатор або змінна).

Якщо в якості константи використовується строчка букв, цифр або інших символів (літерал), то ця строчка заключається в апострофи.

Букви *b*, *q* і *d* можна замінювати виразами xB, xD та xQ відповідно, тоді асемблер не переплутає їх із шістнадцятирічними цифрами.

Наприклад,

1011, 01011xD – десяткові числа;

01011, 1011h – шістнадцяткові числа;

1011q, 1011xQ – вісімкові числа;

01011xB – двійкове число;

«А» – ASCII – код букви А.

Кратні 1024 числа можна задавати за допомогою букви *k*. Число перед *k* – десяткове. Наприклад, 4 Кб можна задати як 4k.

5.7. СПОСОБИ ЗАВДАННЯ ОПЕРАНДІВ КОМАНДИ (СПОСОБИ АДРЕСАЦІЇ)

Розглянемо питання щодо розташування операндів у команді або способи адресації в командах мікропроцесора.

Операнд задається неявно на мікропрограмному рівні. У цьому випадку команда явно не містить операндів. Алгоритм виконання команди використовує деякі об'єкти за умовчанням (реєстри, прапори в реєстрі EFLAGS і т. ін.). Наприклад, команди *cli* і *sti* неявно працюють із прапором переривання IF у реєстрі EFLAGS, а команда *xlat* неявно звертається до реєстра

AL і комірки в пам'яті за адресою, яка визначається вмістом регістрів DS та BX.

Операнд задається в самій команді (безпосередній операнд). Операнд знаходиться в коді команди, тобто є її частиною. Для зберігання такого операнда в команді виділяється поле до 32 бітів. Безпосередній операнд може бути тільки другим операндом (джерелом). Операнд-приймач може знаходитися або в пам'яті, або в регістрі. Наприклад, команда *mov ax,0ffffh* пересилає в регістр AX шістнадцятирічну константу ffff. Команда *add sum,2* складає вміст поля за адресою *sum* з цілим числом 2 і записує результат за місцем першого операнда, тобто в пам'ять.

Операнд знаходиться в одному з регістрів. Регістрові операнди вказують на регістри, у яких зберігається операнд. У цих командах можуть використовуватися наступні регістри:

- 32-розрядні регістри EAX, EBX, ECX, EDX, ESI, EDI, ESP, EBP;
- 16-розрядні регістри AX, BX, CX, DX, SI, DI, SP, BP;
- 8-розрядні регістри AH, AL, BH, BL, CH, CL, DH, DL;
- сегментні регістри CS, DS, SS, ES, FS, GS.

Наприклад, команда *add ax,bx* складає вміст регістрів AX і BX та записує результат у BX. Команда *dec si* зменшує вміст регістра SI на 1.

Операнд розташовується в пам'яті. Це найбільш складний і водночас найбільш гнучкий спосіб завдання операндів. Він дозволяє реалізувати наступні два основні види адресації: пряму і непряму.

У свою чергу, *непряма* адресація має наступні різновиди:

- непряма базова адресація; інша її назва – регістрова непряма адресація;
- непряма базова адресація із зсувом;
- непряма індексна адресація із зсувом;
- непряма базова індексна адресація;
- непряма базова індексна адресація із зсувом.

Операндом є порт введення/виведення. Як ми вже відзначали, крім адресного простору оперативної пам'яті мікропроцесор підтримує адресний простір введення/виведення, який використовується для доступу до пристроїв введення/виведення. Об'єм адресного простору введення/виведення складає 64 Кб. Для будь-якого пристрою комп'ютера в цьому просторі виділяються адреси. Конкретне значення адреси в межах цього простору називається портом введення/виведення. Фізично порту введення/виведення відповідає апаратний регістр (не плутати з регістром мікропроцесора), доступ до якого здійснюється за

допомогою спеціальних команд асемблера *in* і *out*. Наприклад, за допомогою команди *in* al,60h вводиться в регістр AL байт даних з порту 60h.

Регістри, що адресуються за допомогою порту введення/виведення, можуть мати розрядність 8, 16 або 32 біти, але для конкретного порту розрядність регістра фіксована. Команди *in* і *out* працюють із фіксованою номенклатурою об'єктів. Як джерело інформації або одержувача застосовуються регістри-акумулятори EAX, AX, AL. Вибір регістра визначається розрядністю порту. Номер порту може задаватися безпосереднім операндом у командах *in* і *out* або значенням у регістрі DX. Останній спосіб дозволяє динамічно визначити номер порту в програмі.

Наприклад, фрагмент програми може виглядати так:

- *mov* dx,20h; записати номер порту 20h у регістр DX
- *mov* al,20h; записати значення 20h у регістр AL
- *out* dx,al; вивести значення 20h у порт 20H.

Операнд знаходиться в стеку. Команди можуть зовсім не мати операндів, мати один або два операнди. Більшість команд вимагають два операнди, один з яких є операндом-джерелом, а другий – операндом-приймачем. Важливо те, що один операнд може розташовуватися в регістрі або пам'яті, а другий операнд обов'язково повинен знаходитися в регістрі або безпосередньо в команді. Безпосередній операнд може бути тільки операндом-джерелом. У двооперандній машинній команді можливі наступні поєднання операндів:

- регістр – регістр;
- регістр – пам'ять;
- пам'ять – регістр;
- безпосередній операнд – регістр;
- безпосередній операнд – пам'ять.

У даного правила є винятки, які стосуються:

1) команд для роботи з ланцюжками, які можуть переміщати дані з пам'яті в пам'ять;

2) команд для роботи зі стеком, які можуть переносити дані з пам'яті в стек, який також знаходиться в пам'яті;

3) команд типу множення, які окрім операнда, вказаного в команді, використовують ще і другий, неявний операнд.

З перерахованих сполучень операндів найчастіше використовуються регістр - пам'ять і пам'ять – регістр. Розглянемо їх докладніше на прикладах команд асемблера, які показуватимуть, як змінюється формат команди при застосуванні того або іншого виду адресації. У зв'язку з цим зверніть увагу ще раз на рис. 5.8, де показаний принцип формування фізичної адреси

на адресній шині мікропроцесора. Видно, що адреса операнда формується як сума двох складових – зрушеного на 4 біти вмісту сегментного регістра і 16-бітової ефективної адреси, яка в загальному випадку обчислюється як сума трьох компонентів: бази, зміщення і індексу.

Питання для самоперевірки

1. Які мікропроцесори мають архітектуру типу Intel?
2. Наведіть основні етапи еволюції архітектури Intel.
3. У яких режимах працює мікропроцесор типу Intel?
4. Перерахуйте відмінності режимів роботи мікропроцесора.
5. З яких основних блоків регістрів складається програмна модель мікропроцесорів Intel?
6. Наведіть кількісні та якісні характеристики регістрів загального призначення в архітектурі IA-32.
7. Наведіть склад системних регістрів мікропроцесора та їх призначення.
8. Охарактеризуйте системні прапори регістра прапорів мікропроцесора Intel.
9. Що таке технологія MMX та XMM?
10. Охарактеризуйте сегментну модель пам'яті.
11. Наведіть особливості сегментної моделі пам'яті в реальному режимі роботи мікропроцесора.
12. Наведіть особливості сегментної моделі пам'яті в захищеному режимі роботи мікропроцесора.
13. Наведіть механізм формування фізичної адреси в реальному режимі роботи мікропроцесора.
14. Наведіть механізм формування фізичної адреси в захищеному режимі роботи мікропроцесора.
15. Як використовується таблиця дескрипторів при формуванні фізичної адреси пам'яті?
16. Наведіть функціональну класифікацію команд мікропроцесора типу Intel.
17. Охарактеризуйте систему переривань мікропроцесора типу Intel.
18. Наведіть структуру команди мікропроцесора типу Intel.
19. Які способи завдання операндів використовуються в асемблерах мікропроцесорів типу Intel?

РОЗДІЛ 6

АРХІТЕКТУРА ПЕРСОНАЛЬНОГО КОМП'ЮТЕРА

Стрімкий розвиток комп'ютерної техніки і її різноманітного програмного забезпечення – одна з характерних прикмет сучасного періоду розвитку суспільства. Технології, основним компонентом яких є комп'ютер, проникають практично в усі сфери людської діяльності. Те, що нещодавно вважалося фантастикою, тепер завдяки науково-технічному прогресу, особливо прогресу в галузі комп'ютерних технологій, стало реальністю.

6.1. АПАРАТНІ ЗАСОБИ ПЕРСОНАЛЬНОГО КОМП'ЮТЕРА

Персональні комп'ютери, робота яких ґрунтується на принципі програмного керування, мають схожу структуру, спрощений вид якої зображений на рис. 6.1.

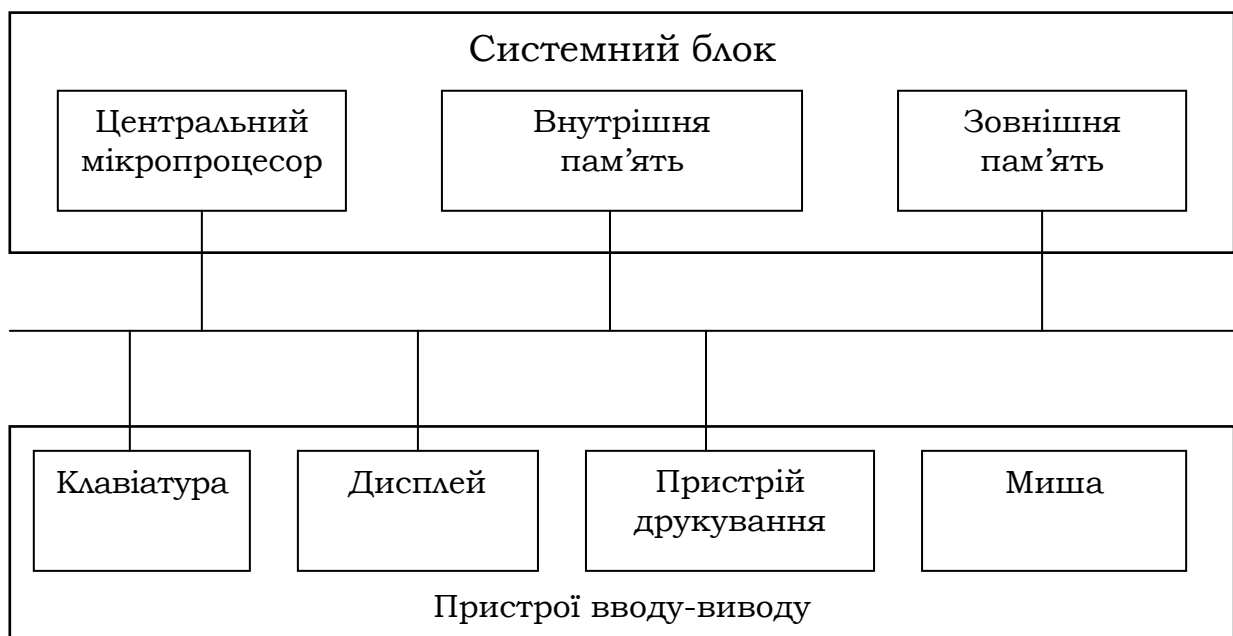


Рис. 6.1. Структура персонального комп'ютера

До апаратних засобів (*Hardware*) відносяться всі пристрої, із яких складається комп'ютер, а також пристрої, які можуть додаватися до нього в міру необхідності. з англійської *Hardware* буквально перекладається «твердий товар» («залізо»): молотки, цвяхи і гвинти. У комп'ютері – це все те, чого можна торкнутися руками, починаючи від сполучних кабелів і закінчуючи монітором, корпусом, клавіатурою і дрібними електронними компонентами (мікросхемами).

До найважливіших складових апаратури персонального комп'ютера (ПК) відносяться:

- системний блок, принаймні з одним дисководом і вінчестером;

– клавіатура, подібна до клавіатури звичайної друкарської машинки, за допомогою якої можна вводити інформацію в комп'ютер;

– мишка;

– дисплей (монітор), на якому відображаються результати процесів, що відбуваються в комп'ютері, результати оброблення інформації.

Додатково до комп'ютера можна підключити, наприклад:

– принтер;

– сканер;

– накопичувач на лазерному диску (CD-ROM);

– звукову плату;

– модем.

6.2. СИСТЕМНИЙ БЛОК

Системна плата

Найважливішим вузлом ПК є системна плата (*main board*), інакше її називають материнською платою (*motherboard*). Системна плата є не у всіх комп'ютерах. У деяких ПК елементи, що звичайно встановлюються на системній платі, розташовані на окремій платі розширення, вставленій в роз'єми системної плати – слоти розширення. У комп'ютерах такого типу плата з роз'ємами називається об'єднувальною платою (*backplane*), а системні блоки подібної конструкції називаються об'єднувальними системними блоками.

Об'єднувальна плата може бути пасивною і активною. На пасивній платі встановлюються роз'єми шини і, можливо, електричні схеми для оброблення буферів і дискових накопичувачів. Уся решта компонентів розташовується на одній або кількох платах розширення, що вставляються в роз'єми об'єднувальної плати. Іноді вся схема розміщається на одній платі розширення, яку називають системною, або материнською картою (*mothercard*). Така системна карта є, по суті, системною платою, що вставляється в роз'єм пасивної об'єднувальної плати. Системи такого типу рідко зустрічаються через високу вартість високопродуктивних системних карт. Конструкції з об'єднувальною платою популярні в промисловості, де їх часто вмонтовують у стійках. Такою ж конструкцією відрізняються деякі могутні сервери.

На активній об'єднувальній платі встановлено контролер шини. Звичайно на ній містяться й інші компоненти. У більшості комп'ютерів на активній об'єднувальній платі розташовуються практично всі вузли звичайної системної плати, окрім процесорного модуля. Процесорний модуль – це плата, на якій встановлені

центральний процесор і всі пов'язані з ним вузли, наприклад, схема синхронізації, кеш-пам'ять і т. ін. Конструкція з процесорним модулем дозволяє легко перевести систему на інший процесор, змінивши всього одну плату. Фактично йдеться про модульну системну плату із замінюваною секцією процесора. У більшості сучасних ПК об'єднувальна плата активна і має окремий процесорний модуль. На жаль, через відсутність стандарту на спосіб взаємодії процесорного модуля з рештою вузлів системи кожна фірма випускає свою плату, яку можна придбати тільки у виробника конкретного комп'ютера. Таке звуження ринку призводить до того, що ця плата дорожча за більшість повної системної плати (з процесором) інших виробників.

Стандартні розміри системної плати

Системна плата випускається в декількох варіантах. Вони відрізняються розмірами, що, у свою чергу, визначає тип корпусу, у якому їх можна встановити. Існують такі основні різновиди системної плати: об'єднувальна плата; повнорозмірна плата AT; Baby-AT; LPX; ATX; NLX.

Об'єднувальна плата

Системна (материнська) плата в повному розумінні цього слова встановлена не у всіх комп'ютерах. У деяких системах ті компоненти, які звичайно знаходяться на системній платі, встановлюються у вже вставлену плату розширення. У таких комп'ютерах головна плата із слотами називається об'єднувальною платою. А комп'ютери, які використовують таку конструкцію, називаються комп'ютерами з об'єднувальною платою.

Системи з об'єднувальною платою бувають двох основних типів: пасивні і активні. Пасивна об'єднувальна плата взагалі не містить ніякої електроніки, окрім хіба що роз'ємів шини і декількох буферів і драйверних схем. Уся решта схем звичайної системної плати розміщена на платі розширення. Є пасивні системи, у яких вся системна електроніка знаходиться на єдиній платі розширення. Практично ця плата є справжньою системною, але вона повинна бути вставлена в слот на пасивній об'єднувальній платі. Така конструкція була розроблена для того, щоб модернізувати систему і мати можливість замінювати в ній будь-яку плату якомога простіше. Але через високу вартість системної плати потрібного типу подібні конструкції дуже рідко зустрічаються в персональних комп'ютерах. А ось у промислових системах пасивна об'єднувальна плата дуже популярна. І ще їх можна зустріти в деяких могутніх серверах.

Активна об'єднувальна плата містить схеми управління шиною і безліч інших компонентів. На більшості такої плати міститься вся електроніка звичайної системної плати, немає тільки

процесорного комплексу. Процесорним комплексом називають ту частину схеми плати, яка включає сам процесор і безпосередньо пов'язані з ним компоненти, такі як тактовий генератор, кеш і т. ін. Виходить, що у вас як би модульна системна плата із замінюваним процесорним комплексом. Більшість сучасних ПК з об'єднувальною платою використовують саме активну плату з окремим процесорним комплексом. Фірми Compaq і IBM використовують таку конструкцію у своїх наймогутніших системах серверного класу. На жаль, інтерфейс процесорних комплексів дотепер не стандартизований.

Обидві конструкції, і з системною платою, і з об'єднувальною, мають свої переваги і недоліки. У кінці 70-х рр. XX ст. у більшості ПК відомих виробників використовувалася об'єднувальна плата. Пізніше за Apple і IBM перейшли до системної плати, оскільки при їх масовому виробництві така конструкція виявилася дешевшою. Проте теоретично перевагою систем з об'єднувальною платою залишається те, що їх легше модернізувати до нового процесора і нового рівня продуктивності, замінюючи тільки невелику другорядну плату. У комп'ютерах із системною платою для заміни процесора часто доводиться міняти всю системну плату, що набагато складніше.

Оперативна пам'ять персонального комп'ютера

Елементи пам'яті разом із чипсетом і центральним процесором складають основу будь-якого персонального комп'ютера, тому що в них зберігаються необхідні для вирішення поставленої задачі дані, що можуть бути в будь-який момент прочитані або змінені.

Пам'ять за всіх часів була критичним ресурсом комп'ютерів. Недарма щоб досягти компромісу між ціною і продуктивністю, в обчислювальних системах вибудовано цілу ієрархію пам'яті, що розрізняється швидкодією. Вона включає зовнішню пам'ять – найдешевшу, але й найповільнішу (яка, до речі, має свою ієрархію), оперативні запам'ятовуючі пристрої (ОЗП) – вони швидші, але дорожчі, кеш-пам'ять – найшвидший, але й найдорожчий ресурс. ОЗП бувають статичні (надоперативні) і динамічні. Статичні приблизно в 13 разів швидші, ніж динамічні (4,5 МГц проти 60 МГц), однак сьогодні вони занадто дорогі й споживають занадто багато енергії, щоб використовувати їх в обсягах, які відповідають вимогам до сучасних ОЗП. Тому застосування статичної пам'яті, як правило, обмежене невеликою за обсягом кеш-пам'яттю першого (Level 1 – L1), другого (L2), третього (L3) чи четвертого (L4) рівнів (якщо вона не інтегрована на один кристал із процесором).

Отже, із конструктивних міркувань у мікросхемах пам'яті деякі групи комірок виявляються зв'язаними. Комірки, об'єднані

керуючим провідником, прийнято називати рядком, чи *row* (немов вони розташовані в квадратній таблиці горизонтально). Комірки ж, об'єднані провідником, що передає значення, називають стовпцем, чи *column* (як ніби вони розташувалися по вертикалі). Таким чином, при виборі рядка зчитуються відразу всі її комірки, тобто на кожному із провідників стовпців виникає напруга, обумовлена логічним значенням відповідної комірки обраного рядка. Описана сукупність комірок і логічні елементи, які їх обрамляють і які пов'язані з вибором рядків і стовпців, і називається ядром інтегральної схеми (ІС). Звичайно, існує ще інтерфейсна логіка, що забезпечує взаємодію ядра з зовнішнім світом. Саме вона в найпростішому випадку відповідає за комутацію того чи іншого стовпця «на вихід» при зчитуванні даних із конкретної комірки. Тут варто розділити поняття логічної організації ІС і структури ядра. Під логічною організацією мають на увазі розрядність мікросхеми (тобто кількість ліній введення/виведення даних) і глибину адресного простору (тобто кількість бітів, що можуть бути збережені для кожного розряду), їх звичайно представляють у вигляді добутку: наприклад, популярні 64-мегабітні ІС випускають з організацією 16 Мбіт $\times$ 4; 8 Мбіт $\times$ 8 і 4 Мбіт $\times$ 16. Добуток, якщо його обчислити, показує обсяг мікросхеми пам'яті (у бітах), часто називаний щільністю (*density*). Таким чином, з погляду логічної організації «рядки» і «стовпці» мають адреси, представлені на зовнішніх висновках, причому кожен «стовпець» містить стільки розрядів, скільки є ліній введення/виведення даних. Існуючі фізичні обмеження описаної організації ядра і засобів доступу до комірок визначають максимально можливу робочу частоту масиву комірок пам'яті. Незважаючи на розвиток напівпровідникових технологій і всі спроби розробників, радикально прискорити роботу ядра не вдається. Тому зусилля творців усіх типів пам'яті, які нещодавно з'явилися, зосередилися на обслуговуючій електроніці. Ускладнивши керуючу та інтерфейсну логіку, можна організувати роботу ядра так, щоб забезпечити обмін даними на багато вищій частоті. Площа, займана логічними елементами, набагато менша площі всього масиву комірок (граничне значення не перевищує 5 %). Тому їхнє ускладнення приводить лише до незначного збільшення площі кристала і, таким чином, майже не позначається на вартості виробництва ІС.

Перед оперативною пам'яттю (RAM – *Random Access Memory*) поставлена задача за вимогою процесора (CPU) надавати будь-яку інформацію, що зберігається в ній, причому робити це

потрібно досить швидко, аби уникнути тривалого простою CPU. Особливістю RAM є те, що вона належить до сімейства динамічної пам'яті, тобто її вміст повинен перезаписуватися через визначений час, інакше інформація буде втрачена. Запам'ятовуючим елементом динамічної пам'яті (будь-якої, не тільки оперативної) є конденсатор, що може знаходитися або в зарядженому, або у розрядженому стані. Уважається, що в першому випадку в комірці пам'яті записана логічна одиниця (1), а в другому логічний нуль (0). В ідеальному конденсаторі заряд повинен зберігатися як завгодно довго. У реальному ж конденсаторі існують втрати потенціалу, і як наслідок записана інформація буде втрачена, тому інформацію треба постійно відновлювати (регенерувати).

Уявити собі функції інтерфейсної логіки в найпростішому випадку можна, якщо розглянути звертання до пам'яті (рис. 6.2). Воно складається з кількох етапів. Спочатку вказується характер майбутньої операції (читання чи запис), потім передається адреса, за якою потрібно зробити обмін даними. Зовнішні (інтерфейсні) адресні лінії у чипів динамічної пам'яті спільні для рядків і стовпців. У зв'язку з тим, що адреси рядків і стовпців не використовуються одночасно, таке об'єднання дозволяє майже без втрат скоротити (ледве не вдвічі) кількість ніжок у IC, а виходить, знизити вартість мікросхеми, адже вона не в останню чергу залежить від кількості ніжок! Таким чином, передання повної адреси відбувається в два етапи: спочатку передається одна адреса (рядка), а потім інша (стовпця). Для стробування кожної частини адреси слугують, відповідно, сигнали RAS (*Row Access Strobe*) і CAS (*Column Access Strobe*; іноді розшифровують інакше – *Column Access Select*), їх активний (стробуючий) рівень звичайно низький. Для забезпечення надійного стробування ці сигнали подаються із затримкою, достатньою для завершення перехідних процесів у ланцюгах, у яких використовуються ці адреси. Оскільки контролер пам'яті, керуючий усім цим процесом, – пристрій синхронний, тобто спрацьовує винятково за тактовими імпульсами, то на кожну операцію потрібен як мінімум такт.

Якщо виконується цикл запису, то подається сигнал WR (*Write*), і інформація надходить на шину стовпця не з регістра, а з інформаційного входу пам'яті через комутатор, визначений адресою стовпця. Таким чином, проходження даних при записі визначається комбінацією сигналів адреси стовпця і рядка і дозволу запису даних у пам'ять. При записі дані з регістра рядка на вихід не надходять.

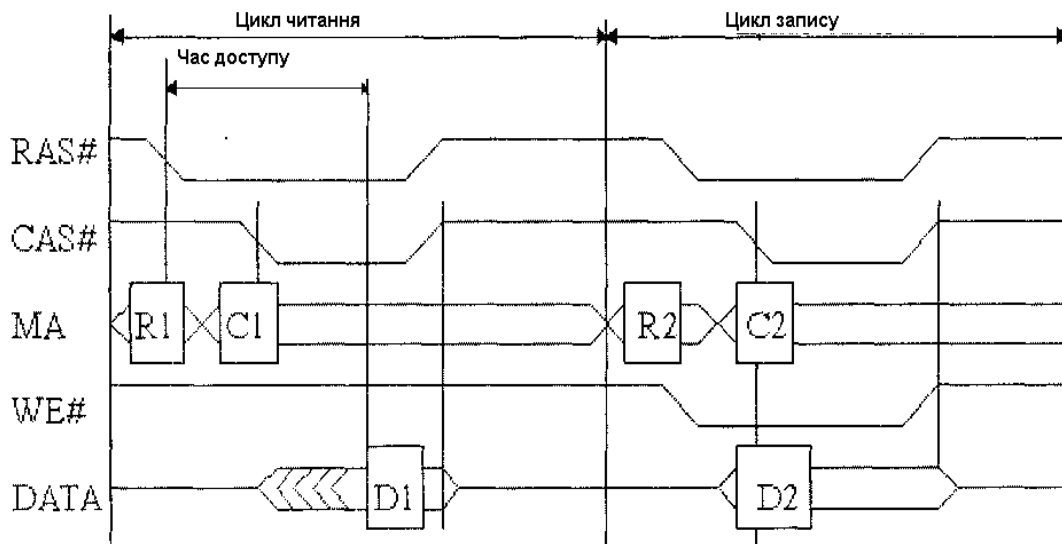


Рис. 6.2. Схема звернення до пам'яті

Важливою характеристикою елементів пам'яті є час доступу, що характеризується інтервалом часу, протягом якого інформація записується в пам'ять чи зчитується з неї. Час доступу в оперативній пам'яті вимірюється в наносекундах. Сьогоднішні мікросхеми пам'яті мають час доступу близько 10 нс. У принципі, на материнську плату можна установити елементи пам'яті з трохи відмінним часом доступу, але це може привести до нестабільної роботи системи чи взагалі до її відсутності. У цьому випадку варто в CMOS Setup (енергетична пам'ять) установити параметри, які відповідають більш повільній пам'яті.

У процесі розвитку комп'ютерної індустрії з'явилися наступні основні типи мікросхем оперативної пам'яті:

1. Динамічна пам'ять (DRAM)

Буква «D» у найменуванні цієї пам'яті говорить про те, що вона динамічна, більшість застосовуваної в ПК пам'яті, у тому числі й оперативної, є динамічною. На дуже старих материнських платах (для CPU 8088 і 80286) мікросхеми DRAM встановлюються окремими DIP-мікросхемами (у виді жука), однак пізніше були розроблені SIP- і SIMM- (однорядні) модулі, що одержали більш широку популярність, у вигляді окремих плат із уже встановленою на них пам'яттю. Згодом SIMM-модулі також витиснули SIP-модулі, тому що ніжки SIP-модулів при їхній установці і вилученні часто обламувалися. На сьогодні DRAM застаріла, тому знайти приклади її мікросхем для більш детального розгляду дуже складно.

2. SDRAM (Synchronous DRAM)

За весь час не було іншої пам'яті, що набула такої великої популярності. Основна відмінність SDRAM від інших видів пам'яті полягає в тому, що всі операції в мікросхемах пам'яті

синхронізовані з тактовою частотою CPU, тобто пам'ять і процесор працюють синхронно. Це досягається шляхом використання внутрішньої триступінчастої конвеєрної архітектури мікросхеми і чергуванням адрес. Технологія SDRAM дозволяє скоротити час, що витрачається на виконання команд і передавання даних, за рахунок виключення циклів очікування. Існують модулі із SDRAM (вони називаються DIMM-модулями і мають 168 контактів, тоді як пам'ять типу EDO/BEDO/FPM DRAM трапляється в основному у вигляді 72-контактних SIMM-модулів), призначені для роботи на частотах 66, 100 і 133 МГц.

Пам'ять може відповідати специфікаціям PC66, PC100 чи PC133, причому перші дві вже не зустрічаються. До кінця 2000 р. для високопродуктивних комп'ютерів було стандартним поєднання Pentium III/133 МГц – PC133 SDRAM, але, коли в 2001 р. почали набирати обертів процесори Pentium IV і AMD Athlon с частотами вище 1 ГГц, стало остаточно ясно, що технологія SDRAM виходить з ужитку. Але в системах категорії Hi-End пам'ять типу SDRAM, яка стала за час свого існування настільки широко розповсюдженою, пальмою першості вже поступилася.

Далі будуть розглянуті дві порівняно нові технології, що претендують на місце в сучасному ПК.

3. RDRAM (*Rambus DRAM*)

Одна з таких технологій – технологія пам'яті RDRAM, що розроблена компанією Rambus і активно просувається на ринок компанією Intel, котра розробляє усі свої останні продукти з розрахунком на цю пам'ять. На відміну від SDRAM, RDRAM використовує більш вузьку (малорозрядну) магістраль даних. Це дозволило в кілька разів підвищити частоту, на якій функціонує пам'ять. Дані передаються двома пакетами за такт. У результаті пропускна здатність RDRAM у багато разів перевищує цей параметр у SDRAM.

4. DDR SDRAM (*Double Data Rate SDRAM*)

Новий тип пам'яті DDR SDRAM з'явився внаслідок покращення архітектури SDRAM, тому інша назва цього типу пам'яті – SDRAM II. Лідерство в розробленні DDR SDRAM належить корпорації Samsung. Пам'ять типу DDR SDRAM може передавати і приймати дані по висхідному і спадному рівню сигналу шини.

6.3. МОНІТОР

Перше, на що обов'язково слід звернути увагу, – ніколи не економте на моніторі. Монітор не можна модернізувати. Він купується один раз для довготривалого використання. Саме

через монітор ми сприймаємо всю візуальну інформацію від комп'ютера. Не важливо, чи працюєте ви з бухгалтерською програмою, пишете листи, граєте в ігри, управляєте сервером – ви завжди використовуєте монітор. Від якості і безпеки монітора безпосередньо залежить ваше здоров'я – перш за все зір.

CRT-монітору (Cathode Ray Tube)

В основі всіх подібних моніторів лежить катодно-променева трубка, але це дослівний переклад, технічно правильно говорити електронно-променева трубка. Розглянемо принципи роботи CRT-моніторів. CRT-монітор має скляну трубку, усередині якої знаходиться вакуум, тобто все повітря видалене. З фронтальної сторони внутрішня частина скла трубки покрита люмінофором (*Luminofor*).

Люмінофор – це речовина, яка випускає світло при бомбардуванні його зарядженими частинками. Для створення зображення в CRT-моніторі використовується електронна гармата, яка випускає потік електронів крізь металеву маску або ґрати на внутрішню поверхню скляного екрана монітора, яка покрита різноколірними люмінофорними точками. Потік електронів на шляху до фронтальної частини трубки проходить через модулятор інтенсивності, який працює за принципом різниці потенціалів. У результаті електрони набувають великої енергії, частина з якої витрачається на свічення люмінофора. Електрони потрапляють на люмінофорний шар, після чого енергія електронів перетворюється на світло, тобто потік електронів примушує точки люмінофора світитися. Ці точки люмінофора, що світяться, формують зображення, яке ви бачите на моніторі. Як правило, у кольоровому CRT-моніторі використовуються три електронні гармати, на відміну від однієї гармати, вживаної в монохромних моніторах, які зараз практично не використовуються і мало кому цікаві.

Усі ми знаємо, що наші очі реагують на основні кольори: червоний (*Red*), зелений (*Green*) і синій (*Blue*) і на їх комбінації, які створюють нескінченну кількість кольорів.

Люмінофорний шар, що покриває фронтальну частину електронно-променевої трубки, складається з дуже маленьких елементів (настільки маленьких, що людське око їх не завжди може розрізнити). Ці люмінофорні елементи відтворюють основні кольори, фактично є три типи різноколірних частинок, чий колір відповідає основним кольорам RGB (звідси і назва групи з люмінофорних елементів – тріади).

Люмінофор починає світитися, як було сказано вище, під впливом прискорених електронів, які створюються трьома електронними гарматами. Кожна з трьох гармат відповідає одному з

основних кольорів і посиляє пучок електронів на різні частинки люмінофорного шару, чие свічення основними кольорами з різною інтенсивністю комбінується, і в результаті формується зображення з необхідним кольором. Наприклад, якщо активувати червону, зелену і синю люмінофорні частинки, то їх комбінація сформує білий колір.

Для управління електронно-променевою трубкою необхідна і управляюча електроніка, якість якої багато в чому визначає і якість монітора.

Електронно-променеві трубки можна розподілити на два класи – трьохпроменеві з дельтаобразним розташуванням електронних гармат і з планарним розташуванням електронних гармат. У цих трубках застосовуються щілисті і тіньові маски, хоча правильніше сказати, що вони всі тіньові. При цьому трубки з планарним розташуванням електронних гармат ще називають кінескопами з самозведенням променів, оскільки дія магнітного поля Землі на три планарно розташованих променя практично однакова і при зміні положення трубки щодо поля Землі не потрібно проводити додаткового регулювання.

Отже, найпоширеніші типи масок – це тіньові, а вони бувають двох типів: *Shadow Mask* (тіньова маска) і *Slot Mask* (щілиста маска).

SHADOW MASK

Тіньова маска – це найпоширеніший тип масок для CRT-моніторів. Тіньова маска складається з металевої сітки перед частиною скляної трубки з люмінофорним шаром. Як правило, більшість сучасних тіньових масок виготовляють з інвару (*invar* – сплав заліза і нікелю). Отвори в металевій сітці працюють як приціл (хоч і не точний), саме цим забезпечується те, що електронний промінь потрапляє тільки на необхідні люмінофорні елементи і лише в певних областях. Тіньова маска створює ґрати з однорідними точками (ще званими тріади), де кожна така точка складається з трьох люмінофорних елементів основних кольорів (зеленого, червоного і синього), – які світяться з різною інтенсивністю під впливом променів з електронних гармат. Зміною струму кожного з трьох електронних променів можна добитися довільного кольору елементу зображення, що утворюється тріадою точок.

Мінімальна відстань між люмінофорними елементами однакового кольору називається *dot pitch* (або крок точки) і є індексом якості зображення. Крок точки звичайно вимірюється в міліметрах. Чим менше значення кроку точки, тим вища якість зображення, яке відтворюється на моніторі.

SLOT MASK

Щілиста маска – це технологія, яка широко застосовується компанією NEC під ім'ям «CromaClear». Це рішення на практиці є комбінацією двох технологій – CRT та Shadow Mask. Е даному випадку люмінофорні елементи розташовані у вертикальних еліптичних осередках, а маска зроблена з вертикальних ліній. Фактично вертикальні смуги розділені на еліптичні осередки, які містять групи з трьох люмінофорних елементів трьох основних кольорів. Мінімальна відстань між двома осередками називається *slot pitch* (щілистий крок). Чим менше значення *slot pitch*, тим вища якість зображення на моніторі.

APERTURE GRILLE

Апертурна решітка (*aperture grill*) – це тип маски, який використовують різні виробники у своїх технологіях для виробництва кінескопів, що по-різному називаються, але мають однакову суть, наприклад технологія, Trinitron від Sony або Diamondtron від Mitsubishi. Це рішення не включає металеві ґрати з отворами, як у випадку з тіньовою маскою, а має ґрати з вертикальних ліній. Замість точок із люмінофорними елементами трьох основних кольорів апертурна решітка містить серію ниток, що складаються з люмінофорних елементів, збудованих у вигляді вертикальних смуг трьох основних кольорів. Така система забезпечує високу контрастність зображення і хорошу насиченість кольорів, що разом забезпечує високу якість моніторів із трубками на основі цієї технології. Маска, яку застосовують в трубках фірми Sony, Mitsubishi, ViewSonic, є тонкою фольгою, на яку нанесені тонкі вертикальні лінії. Вона тримається на горизонтальних (одній у 15-, двох у 17-, трьох і більше у 21-дюймових моніторах) дротинках, тінь від яких ви бачите на екрані. Ці дротинки застосовуються для гасіння коливань та називаються *dampers wire*. Їх добре видно, особливо при світлому фоні зображення на моніторі. Деяким користувачам ці лінії принципово не подобаються, інші ж, навпаки, задоволені і використовують їх як горизонтальну лінійку.

Мінімальна відстань між смугами люмінофора однакового кольору називається *strip pitch* (або кроком смуги) і вимірюється в міліметрах. Чим менше значення *strip pitch*, тим вища якість зображення на моніторі.

Зображення на екрані відтворюється в результаті процесу, у ході якого свічення люмінофорних елементів ініціюється електронним променем, що проходить послідовно по рядках в наступному порядку: зліва направо і зверху вниз на екрані монітора. Цей процес відбувається дуже швидко, тому нам здається, що екран світиться постійно. У сітківці наших очей зображення зберігається близько 1/20 с. Це означає, що якщо

електронний промінь рухатиметься по екрану повільно, ми можемо бачити цей рух як окрему рухому яскраву точку, але коли промінь починає рухатися, швидко прокреслюючи на екрані рядок (хоча б 20 разів за секунду), наші очі не побачать рухомої крапки, а побачать лише рівномірну лінію на екрані. Якщо тепер примусити промінь послідовно пробігати по багатьох горизонтальних лініях зверху вниз за час менший $1/25$ с, ми побачимо рівномірно освітлений екран з невеликим мерехтінням. Рух самого променя відбуватиметься настільки швидко, що наше око не буде в змозі його помітити. Чим швидше електронний промінь проходить по всьому екрану, тим менше буде помітне і мерехтіння картинки. Уважається, що таке мерехтіння стає практично непомітним при частоті повторення кадрів (проходів променя по всіх елементах зображення) приблизно 75 за секунду. Проте ця величина деякою мірою залежить від розміру монітора. Справа в тому, що периферійні області сітківки ока містять світлочутливі елементи з меншою інерційністю. Тому мерехтіння моніторів з великими кутами огляду стає помітним при великих частотах кадрів. Здатність електроніки формувати на екрані дрібні елементи зображення залежить від ширини смуги пропускання (*bandwidth*). Ширина смуги пропускання монітора пропорційна числу пікселів, із яких формує зображення відеокарта вашого комп'ютера. До ширини смуги пропускання монітора ми ще повернемося. Тепер перейдемо до іншого типу моніторів – LCD.

LCD Monitors

LCD (*Liquid Crystal Display* – рідкокристалічні монітори) зроблені з речовини, яка знаходиться в рідкому стані, але при цьому має деякі властивості, притаманні кристалічним тілам. Рідкі кристали були відкриті досить давно, але спочатку вони використовувалися з іншою метою. Молекули рідких кристалів під впливом електрики можуть змінювати свою орієнтацію і внаслідок цього змінювати властивості світлового променя того, що проходить крізь них. Ґрунтуючись на цьому відкритті і в результаті подальших досліджень стало можливим виявити зв'язок між підвищенням електричної напруги і зміною орієнтації молекул кристалів для забезпечення створення зображення. Перше своє застосування рідкі кристали знайшли в дисплеях для калькуляторів і у кварцовому годиннику, а потім їх стали використовувати в моніторах для портативних комп'ютерів. Сьогодні в результаті прогресу в цій галузі починають набувати все більшого поширення LCD-монітори для настільних комп'ютерів. Далі мова піде тільки про традиційні LCD-монітори, так звані Nematic LCD.

Екран LCD-монітора є масивом маленьких сегментів (пікселів), які можуть маніпулюватися для відображення інформації. LCD-монітор має декілька шарів, де ключову роль грають дві панелі, зроблені з вільного від натрію і дуже чистого скляного матеріалу, званого субстрат, або підкладка, які, власне, і містять тонкий шар рідких кристалів між собою. На панелях є борозенки, які направляють кристали, повідомляючи їм спеціальну орієнтацію. Борозенки розташовані таким чином, що вони паралельні на кожній панелі, але перпендикулярні між двома панелями. Вертикальні борозенки виходять у результаті розміщення на скляній поверхні тонких плівок із прозорого пластика, який потім спеціальним чином обробляється. Стикаючись із борозенками, молекули в рідких кристалах орієнтуються однаково у всіх осередках. Молекули одного з різновидів рідких кристалів за відсутності напруги повертають вектор електричного (і магнітного) поля в такій світловій хвилі на деякий кут у площині, перпендикулярній осі розповсюдження пучка. Дві панелі розташовані дуже близько одна до одної. Рідкокристалічна панель освітлюється джерелом світла (залежно від того, де він розташований, рідкокристалічні панелі працюють на віддзеркалення або на проходження світла). Площина поляризації світлового променя повертається на 90° при проходженні однієї панелі.

При появі електричного поля молекули рідких кристалів частково шикуються уздовж поля і кут повороту площини поляризації світла стає відмінним від 90° .

Для виведення кольорового зображення необхідне підсвічування монітора ззаду так, щоб світло породжувалося в задній частині LCD-дисплею. Це необхідно для того, щоб можна було спостерігати зображення з хорошою якістю. Колір виходить у результаті використання трьох фільтрів, які виділяють з випромінювання джерела білого світу три основні компоненти. Комбінуючи три основні кольори для кожної точки або пікселя екрану, з'являється можливість відтворити будь-який колір.

Перші LCD-дисплеї були дуже маленькими, близько 8 дюймів, тоді як сьогодні вони досягли 15 дюймів у ноутбуках, а для настільних комп'ютерів – 17, 19 і більше. Услід за збільшенням розмірів слідує збільшення роздільної здатності екрана, наслідком чого є поява нових проблем, які були вирішені за допомогою спеціальних технологій, що з'явилися нещодавно. Однією з перших проблем була необхідність стандарту у визначенні якості відображення при високій роздільній здатності екрана. Першим кроком на шляху до мети було збільшення кута повороту площини поляризації світла в кристалах із 90° до 270° .

Коротко розповімо про роздільну здатність LCD-моніторів. Її ще називають *native*, вона відповідає максимальному фізичному

дозволу CRT-моніторів. Саме в *native* роздільної здатності LCD-монітор відтворює зображення краще за все. Ця роздільна здатність визначається розміром пікселів, який у LCD-монітора фіксований. Наприклад, якщо LCD-монітор має *native* дозвіл 1024×768, то це означає, що на кожній з 768 ліній розташовано 1024 електродів (читай – пікселів). При цьому є можливість використовувати і нижче, ніж *native*, роздільну здатність. Для цього є два способи. Перший називається *Centering* (центрування). Суть методу в тому, що для відображення зображення використовується тільки та кількість пікселів, яка необхідна для формування зображення з нижчою роздільною здатністю. У результаті зображення виходить не у весь екран, а тільки у середині. Усі невживані пікселі залишаються чорними, тобто навколо зображення утворюється широка чорна рамка. Сутність другого методу полягає в тому, що при відтворенні зображення з нижчою, ніж *native*, роздільною здатністю використовуються всі пікселі, тобто зображення займає весь екран. Проте через те що зображення розтягується на весь екран, виникають невеликі спотворення і погіршується різкість. Тому при виборі LCD-монітора важливо чітко знати, яка саме роздільна здатність вам потрібна.

Окремо варто згадати про яскравість LCD-моніторів, оскільки поки немає ніяких стандартів для визначення того, чи достатню яскравість має LCD-монітор. При цьому в центрі яскравість LCD-монітора може бути на 25 % вища, ніж біля країв екрана. Єдиний спосіб визначити, чи підходить вам яскравість конкретного LCD-монітора, це порівняти його яскравість з іншими LCD-моніторами.

І останній параметр, про який потрібно згадати, це контрастність. Контрастність LCD-монітора визначається відношенням яскравостей між найяскравішим білим і найтемнішим чорним кольором. Хорошим контрастним співвідношенням вважається 120 : 1, що забезпечує відтворення живих насичених кольорів. Контрастне співвідношення 300 : 1 і вище використовується тоді, коли потрібне точне відображення чорно-білих півтонів. Але, як і у випадку з яскравістю, поки що немає ніяких стандартів, тому головним, визначальним чинником є ваші очі.

Варто відзначити і таку особливість частини LCD-моніторів, як можливість повороту самого екрану на 90° з одночасним автоматичним переверненням зображення. Це, зокрема, дає можливість лист A4 формату повністю умістити на екрані, що дуже зручно під час верстання. Правда, серед CRT-моніторів теж є моделі з такою можливістю, але вони украй рідкісні. У випадку з LCD-моніторами ця функція стає майже стандартною.

До переваг LCD-моніторів можна віднести те, що вони дійсно плоскі в буквальному розумінні цього слова, а створюване на їх екранах зображення відрізняється чіткістю і насиченістю кольорів. Відсутність спотворень на екрані і маси інших проблем, властивих традиційним CRT-моніторам, також є особливістю LCD-моніторів. Додамо, що споживана і розсіююча потужність у LCD-моніторів істотно нижча, ніж у CRT-моніторів.

Головною проблемою розвитку технологій LCD для сектора настільних комп'ютерів, схоже, є розмір монітора, який впливає на його вартість. Із зростанням розмірів дисплеїв знижуються виробничі можливості.

Plasma monitors

Такі виробники, як Fujitsu, Matsushita, Mitsubishi, NEC, Pioneer та ін., уже почали виробництво плазмових моніторів із діагоналлю 40 дюймів, причому деякі моделі вже готові для масового виробництва. Робота плазмових моніторів дуже схожа на роботу неонових ламп, які зроблені у вигляді трубки, заповненої інертним газом низького тиску. Всередину трубки поміщена пара електродів, між якими запалюється електричний розряд і виникає свічення.

Плазмові екрани створюються шляхом заповнення простору між двома скляними поверхнями інертним газом, наприклад аргоном або неоном. Потім на скляну поверхню поміщають маленькі прозорі електроди, на які подається високочастотна напруга. Під дією цієї напруги в прилеглій до електроду газовій області виникає електричний розряд. Плазма газового розряду випромінює світло в ультрафіолетовому діапазоні (видимому людиною) та викликає свічення частинок люмінофора. Фактично кожен піксель на екрані працює як звичайна флуоресцентна лампа (інакше кажучи, лампа денного світла). Висока яскравість і контрастність разом із відсутністю тремтіння є великими перевагами таких моніторів. Крім того, по відношенню до нормалі кут, під яким можна побачити нормальне зображення на плазмових моніторах, істотно більший, ніж 45° (у порівнянні з LCD-моніторами). Головними недоліками такого типу моніторів є досить висока споживана потужність, що зростає при збільшенні діагоналі монітора, і низька роздільна здатність, обумовлена великим розміром елементу зображення. Окрім цього, властивості люмінофорних елементів швидко погіршуються, і екран стає менш яскравим, тому термін служби плазмових моніторів обмежений 10 тис. год (це близько 5 років при офісному використанні). Через ці обмеження, такі монітори використовуються поки тільки для конференцій, презентацій, інформаційних щитів, тобто там, де потрібні великі розміри екранів для відображення інформації.

Проте є всі підстави припускати, що незабаром існуючі технологічні обмеження будуть подолані, а при зниженні вартості такий тип пристроїв може з успіхом застосовуватися як телевізійні екрани або монітори для комп'ютерів. Подібні телевізори вже є, вони мають велику діагональ, дуже тонкі, однак коштують дуже дорого.

Ряд провідних розробників у галузі LCD і Plasma екранів спільно розробляють технологію PALC (*Plasma Addressed Liquid Crystal*), яка повинна поєднати в собі переваги плазмових і LCD-екранів з активною матрицею.

FED-монітори

Технології, які застосовуються при створенні моніторів, можуть бути розділені на дві групи:

1) монітори, засновані на випромінюванні світла, наприклад традиційні CRT-монітори і плазмові, тобто це пристрої, елементи екрана яких випромінюють світло в зовнішній світ;

2) монітори типу трансляції, такі як LCD-монітори.

Одним із кращих технологічних напрямів у галузі створення моніторів, який суміщає в собі особливості обох технологій, описаних нами вище, є технологія FED (*Field Emission Display*). Монітори FED засновані на процесі, який трохи схожий на той, що застосовується в CRT-моніторах, оскільки в обох методах застосовується люмінофор, що світиться під впливом електронного променя. Головна відмінність між CRT- і FED-моніторами полягає в тому, що CRT-монітори мають три гармати, які випускають три електронні промені, послідовно скануючи панель, покриту люмінофорним шаром, а у FED-моніторі використовуються безліч маленьких джерел електронів, розташованих за кожним елементом екрана, і всі вони розміщуються в просторі по меншій глибині, ніж потрібно для CRT. Кожне джерело електронів управляється окремим електронним елементом, так само як це відбувається в LCD-моніторах, і кожен піксель потім випромінює світло завдяки дії електронів на люмінофорні елементи, як і в традиційних CRT-моніторах. При цьому FED-монітори дуже тонкі.

LEP-монітори

Протягом останніх 30 років увага багатьох учених була звернена на полімерні матеріали (простіше кажучи на пластики), що володіють властивостями провідності і напівпровідності.

Найцікавішим застосуванням пластикових напівпровідників на даний момент є створення різного роду пристроїв відображення інформації на їх базі.

Про те, що напівпровідний пластик під дією електричного струму може випускати фотони (тобто світитися), знали давно. Але украй низька (0,01 %) квантова ефективність цього процесу

(відношення числа випущених фотонів до пропущених через пластик зарядів) робила практичне застосування цього ефекту неможливим. За останній час компанія CDT зробила прорив у цьому напрямі, довівши квантову ефективність двохслойного пластика до 5 % при випромінюванні жовтого світла, що порівняно з ефективністю сучасних неорганічних світлодіодів (LED). Крім підвищення ефективності, вдалося розширити і спектр випромінювання. Тепер пластик може випускати світло в діапазоні від синього до ближнього інфрачервоного з ефективністю близько 1 %.

На сьогодні компанія може представити монохромні (жовтого свічення) LEP-дисплеї, що наближаються за ефективністю до рідкокристалічних дисплеїв LCD (*Liquid Crystal Display*), що поступаються їм за терміном служби, але мають ряд істотних переваг. Оскільки багато стадій процесу виробництва LEP-дисплеїв співпадають з аналогічними стадіями виробництва LCD, виробництво легко переобладнати. Крім того, технологія LEP дозволяє наносити пластик на гнучку підкладку великої площі, що неможливе для неорганічного світлодіода (там доводиться використовувати матрицю діодів).

Оскільки пластик сам випромінює світло, не потрібні підсвічування та інші хитрощі, необхідні для отримання кольорового зображення на LCD-моніторі. Оскільки LEP-дисплей працює при низькій напрузі живлення (менше 3 В) і має малу вагу, його можна використовувати в портативних пристроях, що працюють від батарей.

LEP-дисплей має малий час перемикання (менше 1 мікросекунди), тому його можна використовувати для відтворення відеоінформації. Оскільки шар пластика дуже тонкий, можна використовувати спеціальні покриття для досягнення високої контрастності зображення навіть при сильному зовнішньому засвіченні.

Відеоадаптери

В оригінальній моделі IBM PC на екрані монітора могла відображатися тільки алфавітно-цифрова інформація. Перший відеоадаптер називався *Monochrome Display and Parallel Printer Adapter* (MDPPA), або MDA. Роздільна здатність адаптера MDA дозволяла відображати на моніторі 720 точок по ширині і 350 точок (пікселів) по висоті екрана. Графічного режиму в адаптері передбачено не було, а алфавітно-цифрова інформація відображалася на екрані в 25 рядків по 80 символів у кожному.

За декілька місяців після випуску першої моделі PC з MDA фірма IBM розробила відеоадаптер, який підтримував не тільки графічне зображення, але й кольори, що, до речі, особливо

підкреслювалося навіть у його назві. Адаптер CGA (*Color Graphics Adapter*) забезпечував відображення чотирьох кольорів при роздільній здатності 320×200 пікселів. Трохи пізніше стало зрозуміло, що графіка на CGA, навіть кольорова, не завжди задовольняє вирішуваним задачам, зокрема, через низьку роздільну здатність. Перший відеоадаптер для IBM PC, що якоюсь мірою відповідав цим потребам, був створений на фірмі Hercules у 1982 р. Адаптер HGC (*Hercules Graphics Card*) підтримував на монохромному моніторі роздільну здатність 720×350 точок.

Новою розробкою фірми IBM став поліпшений графічний адаптер EGA (*Enhanced Graphics Adapter*), який з'явився на світ вже в 1984 р. Цей адаптер не тільки дозволяв повністю емулювати всі режими робіт попередніх адаптерів (MDA, CGA), але й, зрозуміло, мав інші додаткові можливості. Наприклад, при роздільній здатності 640×350 пікселів він міг одночасно відтворювати 16 кольорів з палітри в 64 кольори (саме для цього адаптера використовувалися сигнали RrGgBb).

Відеоадаптер VGA (*Video Graphics Array*) був представлений фірмою IBM ще в 1987 р., і повністю був сумісний з адаптером EGA, що забезпечило спадкоємність існуючого програмного забезпечення. Незабаром VGA став фактичним стандартом, що включає всі режими попередніх адаптерів і що розширює їх можливості за роздільною здатністю і кількістю відтворних кольорів. Так, при використанні адаптера VGA забезпечується роздільна здатність 640×480 пікселів, і на екрані монітора може відтворюватися 16 кольорів. При роздільній здатності 320×200 відеоадаптер VGA відтворював 256 кольорів – популярний режим ігрових програм.

Усі режими VGA, виключаючи графічні з роздільною здатністю 640×480 пікселів, використовують вертикальну розгортку з частотою 70 Гц, що істотно знижує мерехтіння екрана, що відчувається користувачем. Частота розгортки для режиму 640×480 точок складає тільки 60 Гц. Основними вузлами VGA-адаптера є власне відеоконтролер (як правило, замовлена BIC-ASIC), відео-BIOS, відео пам'ять, спеціальний цифро-аналоговий перетворювач із невеликою власною пам'яттю (RAMDAC – *Random Access Memory Digital to Analog Converter*), кварцовий осцилятор (один або декілька) і мікросхеми інтерфейсу з системною шиною.

Після того як стало ясно, що стандарт VGA практично повністю себе вичерпав, більшість незалежних розробників почали його покращувати як за рахунок збільшення роздільної спроможності і кількості відтворних кольорів, так і введення нових додаткових можливостей. Хоча всі виробники забезпечували

сумісність своїх виробів з VGA, за додатковими відео режимами і можливостями адаптери часто не співпадали, оскільки кожен вважав за потрібне робити це по-своєму.

Зрозуміло, що вже саме поняття SVGA, не пов'язане жорстко з конкретними режимами роботи адаптера, вносило серйозну плутанину.

Асоціація VESA запропонувала свій стандарт на нові відео адаптери. Спочатку VESA рекомендувала використовувати режим із дозволом 800×600 точок і підтримкою 16 кольорів як стандартний. Наступні – 256-кольорові режими з дозволом 640×480 , 800×600 і 1024×768 точок, а також 16-кольоровий режим з дозволом 1024×768 пікселів і т. д.

Сучасні відео адаптери дозволяють використовувати режим 1024×768 і вище, використовуючи при цьому 24 і 32-бітовий колір (*TrueColor*). Для цього вони володіють великим об'ємом відео пам'яті від 4 до 16 Мб, а також підтримують специфікацію 3Dfx, що дозволяє швидше відтворювати колірні спецефекти.

На величину максимально підтримуючу монітором роздільну здатність безпосередньо впливає частота горизонтальної розгортки електронного променя, вимірювана в кілогерцах (кГц). Значення горизонтальної розгортки монітора показує, яку граничну кількість горизонтальних рядків на екрані монітора може прокреслити електронний промінь за одну секунду. Відповідно, чим вище це значення (а саме воно, як правило, указується на коробці для монітора), тим вищу роздільну здатність може підтримувати монітор при прийнятній частоті кадрів. Гранична частота рядків є критичним параметром при розробці CRT-монітора. У таких моніторах використовуються магнітні системи відхилення електронного променя, що є обмотками з досить великою індуктивністю. Амплітуда імпульсів перенапруження на котушках рядкової розгортки зростає з частотою рядків, тому цей вузол виявляється одним із найбільш напружених місць конструкції і одним з головних джерел перешкод у широкому діапазоні частот. Потужність, споживана вузлами рядкової розгортки, також є одним із серйозних чинників моніторів, що враховуються при проектуванні.

Частота регенерації, або оновлення (кадрової розгортки для CRT-моніторів) екрана, – це параметр, що визначає, як часто зображення на екрані наново перемальовувалося. Частота регенерації вимірюється в герцах (Гц), де 1 Гц відповідає одному циклу в секунду. Наприклад, частота регенерації монітора в 100 Гц означає, що зображення оновлюється 100 разів за секунду. Мерехтіння зображення (*flicker*) приводить до стомлення очей, головних болів і навіть до погіршення зору. Відмітимо, що

чим більший екран монітора, тим більше помітно мерехтіння, особливо периферійним (бічним) зором, оскільки кут огляду зображення збільшується. Значення частоти регенерації залежить від використовуваної роздільної здатності, від електричних параметрів монітора і від можливостей відеоадаптера. Мінімумом безпечною частотою кадрів вважається 75 Гц, при цьому існують стандарти, що визначають значення мінімально допустимої частоти регенерації. Уважається, що чим вище значення частоти регенерації, тим краще, проте дослідження показали, що при частоті вертикальної розгортки вище 110 Гц очі людини вже не можуть помітити ніякого мерехтіння. Нижче ми наводимо таблицю з мінімально допустимими частотами регенерації моніторів за стандартом ТСО'99 для різної роздільної здатності:

Таблиця 6.1

Мінімальні параметри частоти регенерації

Діагональ монітора	Частота регенерації	Роздільна здатність
14–15"	≥ 85 Гц	≥ 800 × 600
17"	≥ 85 Гц	≥ 1024 × 768
19–21"	≥ 85 Гц	≥ 1280 × 1024
> 21"	≥ 85 Гц	≥ 1280 × 1024

Слід зауважити, що в таблиці 6.1 наведені мінімально допустимі параметри, а рекомендована частота регенерації ≥ 100 Гц.

Щоб дізнатися налаштування свого монітора, необхідно відкрити «Панель управління – Екран».

Перейдемо до питання про стандарти безпеки. На сучасних моніторах можна зустріти наклейки з аббревіатурою ТСО або MPRII. На дуже старих моделях зустрічаються ще й написи *Low Radiation*, які насправді ні про що не говорять. Просто колись, виключно в маркетингових цілях, виробники з Південно-Східної Азії привертали цим увагу до своєї продукції. Ніякого захисту подібний напис не гарантує.

Сертифікати ТСО і MPRII

Усі ми хоч раз чули про те, що монітори небезпечні для здоров'я. З метою зниження ризику для здоров'я різними організаціями були розроблені рекомендації з параметрів моніторів, слідуючи яким виробники моніторів борються за наше здоров'я. Усі стандарти безпеки для моніторів регламентують максимально допустимі значення електричних і магнітних полів, створюваних монітором при роботі. Практично в кожній розвиненій країні є власні стандарти, але особливу популярність у всьому світі (так склалося історично) завоювали стандарти, розроблені в Швеції і відомі під іменами ТСО і MPRII. Розповімо про них докладніше.

ТСО

«ТСО (*The Swedish Confederation of Professional Employees* – Шведська конфедерація професійних колективів робочих), членами якої є 1,3 млн шведських професіоналів, організаційно складається з 19 об'єднань, які працюють разом з метою поліпшення умов роботи своїх членів. Ці 1,3 млн членів представляють широкий спектр робочих і службовців із державного і приватного сектора економіки.

«Учителі, інженери, економісти, секретарі і няньки – лише частка з груп, які всі разом формують ТСО. Це означає, що ТСО відображає великий зріз суспільства, що забезпечує їй широку підтримку».

Це цитата з офіційного документа ТСО. Справа в тому, що більше 80 % службовців і робочих у Швеції працюють з комп'ютерами, тому головне завдання ТСО – розробити стандарти безпеки при роботі з комп'ютерами, тобто забезпечити своїм членам і всім іншим безпечно і комфортно робоче місце. Окрім розроблення стандартів безпеки, ТСО бере участь у створенні спеціальних інструментів для тестування моніторів і комп'ютерів.

Стандарти ТСО розроблені для гарантування користувачам комп'ютерів безпечної роботи. Цим стандартам повинен відповідати кожен монітор, що продається у Швеції і Європі. Рекомендації ТСО використовуються виробниками моніторів для створення якісніших продуктів, які менш небезпечні для здоров'я користувачів. Суть рекомендацій ТСО полягає не тільки у визначенні допустимих значень різного типу випромінювань, але й у визначенні мінімально прийнятних параметрів моніторів, наприклад підтримувана роздільна здатність, інтенсивність свічення люмінофора, запас яскравості, енергоспоживання, шумність і т. ін. Більше того, окрім вимог, у документах ТСО наводяться докладні методики тестування моніторів. Деякі документи і додаткову інформацію можна знайти на офіційному сайті ТСО: <http://www.tco-info.com>.

До складу розроблених ТСО рекомендацій сьогодні входять три стандарти: ТСО'92, ТСО'95 і ТСО'99 (неважко здогадатися, що цифри означають рік їх ухвалення).

Більшість вимірювань під час тестувань на відповідність стандартам ТСО проводяться на відстані 30 см від екрана, і на відстані 50 см навколо монітора. Для порівняння під час тестування моніторів на відповідність іншому стандарту (MPRII) усі вимірювання проводяться на відстані 50 см від екрана і навколо монітора. Це пояснює те, що стандарти ТСО жорсткіші, ніж MPRII.

ТСО'92

Стандарт ТСО'92 був розроблений виключно для моніторів і визначає величину максимально допустимих електромагнітних випромінювань при роботі монітора, і так само встановлює стандарт на функції енергозбереження моніторів. Крім того, монітор, сертифікований ТСО'92, повинен відповідати стандарту на енергоспоживання NUTEK і відповідати Європейським стандартам на пожежну і електричну безпеку.

ТСО'95

Стандарт ТСО'95 поширюється на весь персональний комп'ютер, тобто на монітор, системний блок і клавіатуру, і стосується ергономічних властивостей, випромінювань (електричних і магнітних полів, шуму і тепла), режимів енергозбереження і екології (з вимогою до обов'язкової адаптації продукту і технологічного процесу виробництва на фабриці). Відмітимо, що в даному випадку термін «персональний комп'ютер» включає робочі станції, сервери, настільні і підлогові комп'ютери, а також комп'ютери Macintosh.

Стандарт ТСО'95 існує разом із ТСО'92 і не відміняє останній.

ТСО'99

ТСО'99 пред'являє жорсткіші вимоги, ніж ТСО'95, у наступних галузях: ергономіка (фізична, візуальна і зручність використання), енергія, випромінювання (електричних і магнітних полів), навколишнє середовище і екологія, а також пожежна і електрична безпека. Стандарт ТСО'99 поширюється на традиційні CRT-монітори, плоскопанельні монітори (*Flat Panel Displays*), портативні комп'ютери (*Laptop* і *Notebook*), системні блоки і клавіатури.

У розробці стандарту ТСО'99 взяли участь TCO, Naturskyddsforeningen і Statens Energimyndighet (The Swedish National Energy Administration – Шведське національне агентство з енергетики).

Екологічні вимоги включають обмеження на присутність важких металів, бромінатів і хлоринатів, фреонів (CFC) і хлорованих речовин усередині матеріалів.

Будь-який продукт повинен бути підготовлений до перероблення, а виробник зобов'язаний мати розроблену політику з утилізації, яка повинна виконуватися в кожній країні, у якій діє компанія.

Вимоги з енергозбереження включають необхідність того, щоб комп'ютер і/або монітор після певного часу бездіяльності знижували рівень споживання енергії на одну або більше ступенів. При цьому період часу відновлення до робочого режиму споживання енергії повинен влаштовувати користувача.

MPRII

Це ще один стандарт, розроблений у Швеції, де уряд і неурядові організації дуже сильно піклуються про здоров'я населення країни. MPRII був розроблений *The Swedish Board for Technical Accreditation*. Він визначає максимально допустимі величини випромінювання магнітного і електричного полів, а також методи їх вимірювання. MPRII базується на концепції про те, що люди живуть і працюють у місцях, де вже є магнітні і електричні поля, тому пристрої, які ми використовуємо, такі як монітор для комп'ютера, не повинні створювати електричні і магнітні поля, більші ніж ті, які вже існують. Відмітимо, що стандарти ТСО вимагають зниження випромінювань електричних і магнітних полів від пристроїв настільки, наскільки це технічно можливо, незалежно від електричних і магнітних полів, що вже існують навколо нас. Утім, ми вже відзначали, що стандарти ТСО жорсткіші, ніж MPRII.

6.4. ТЕХНОЛОГІЇ ДРУКУЮЧИХ ПРИСТРОЇВ

Персональний комп'ютер є цілком самостійним пристроєм, у якому є все необхідне для автономної роботи. Однак сьогодні важко уявити комп'ютер без принтера.

Сьогодні користувачів комп'ютерів, як правило, хвилює вже не тільки питання, яку саме модель принтера придбати, але й не менш важливі проблеми, пов'язані, наприклад, з постійною наявністю витратних матеріалів у фірми-продавця, можливістю використання кирилических шрифтів, подальшим сервісним обслуговуванням друкуючих пристроїв і т. ін.

Особливу увагу приділимо технології кольорових друкуючих пристроїв. Відзначимо, що виведення монохромного зображення на цих пристроях також можливе. Але перш ніж безпосередньо перейти до опису технологій кольорових принтерів, коротко нагадаємо принцип роботи моніторів.

Як відомо, більшість сучасних настільних комп'ютерів використовують монітори на базі електронно-променевих трубок (ЕПТ). За принципом дії подібні монітори мало чим відрізняються від звичайного телевізора: пучок електронів, що випускається електродом (електронною гарматою), потрапляючи на екран, покритий люмінофором, викликає його свічення. Відмітимо, що будь-яке текстове або графічне зображення на екрані монітора комп'ютера (як, утім, і телевізора) складається з безлічі дискретних точок люмінофора, що іменуються також пікселями (*pixel – picture element*). Тому такі дисплеї називають ще растровими. У кольорового монітора є вже три електронні гармати з окремими схемами управління, а на поверхні екрана

нанесений люмінофор трьох основних кольорів: R (red – червоний), G (green – зелений), B (blue – синій). Ці кольори називаються звичайно первинними, оскільки шляхом складання відповідної їх кількості можна одержати будь-який інший колір. Тому така модель кольороутворення називається адитивною (*addition* – складання), або RGB.

Принтери, здатні виводити графічну інформацію, є, взагалі кажучи (так само як і згадувані вище монітори), растровими пристроями. Проте працюють вони вже з іншими первинними кольорами і використовують, відповідно, іншу модель кольороутворення – субтрактивну (*subtraction* – віднімання). Це може створювати великі проблеми при виведенні інформації з екрана на принтер, оскільки не завжди досягається повна відповідність кольорів. Для цього звичайно слугує спеціальне програмне забезпечення. Проте даної проблеми ми поки не торкатимемося.

Отже, первинними кольорами для кольорових принтерів є зелено-блакитний (*cyan*), світло-червоний (*magenta*) і жовтий (*yellow*). Накладання двох із цих первинних кольорів у даному випадку дає червоний, зелений або блакитний колір. Змішення всіх трьох первинних кольорів субтрактивної моделі дає чорний колір. У деяких принтерах для отримання істинно чорного кольору використовується окремий чорний фарбник (*black*), тому дана модель кольороутворення називається також СМУ (або СМУК).

Пояснимо, чому, власне, розрізняються моделі кольороутворення для моніторів і принтерів. Нагадаємо, що наші очі є складною оптичною системою, яка сприймає випромінюване або вже відображене від освітлюваних предметів світло, зрозуміло, якщо вони самі його не випромінюють. Колір, як відомо, визначається довжиною хвилі електромагнітного випромінювання, певний частотний спектр якого і представляє для нас видиме світло. Тепер неважко зрозуміти, що нанесені на екран точки люмінофора сприймаються такого кольору, який вони і випромінюють. Фарбник же, нанесений на папір, навпаки, діє як фільтр, поглинаючи (віднімаючи!) одні і відображаючи інші довжини електромагнітних хвиль. Нагадаємо також, що насиченість кольору (рожевий, червоний, пурпурний) залежить від кількості білого кольору. Таким чином, проміжні кольори при виведенні зображення, наприклад, рожевого, виходять, як правило, шляхом пропуску (не друку) декількох точок.

Власне, це звичайний підхід, пов'язаний із растрованням зображення (*dithering*). Тобто в цьому випадку відтінки відповідного кольору виходять шляхом угруповання декількох точок зображення в псевдопікселі розміром 2×2, 3×3 і більше точок.

Відношення кількості кольорових точок до білих і визначає рівень насиченості кольору.

Слід також звернути увагу на папір, який використовується для кольорових друкуючих пристроїв. Багато фірм – виробників кольорових принтерів декларують можливість друку на звичайному гладкому (або рівному) папері – *plain paper*. Мабуть, не існує однозначної відповіді на питання, що це за папір. Зрозуміло, багато що залежить від його якості, ступеня оброблення, ваги і навіть кольору. У багатьох користувачів даний вид паперу асоціюється зі звичайним офісним папером загального призначення, який часто використовується для ксерокопіювання. Проте за винятком тільки одного-двох типів принтерів, що використовують спеціальну технологію (про це трохи нижче), усі інші для якісного виведення зображення вимагають спеціального офісного (конторського) паперу або спеціального паперу для принтерів. Останній вид паперу звичайно використовується при монохромному лазерному друці. Збільшений рельєф поверхні цих трьох видів паперу наведений на рис. 6.3.

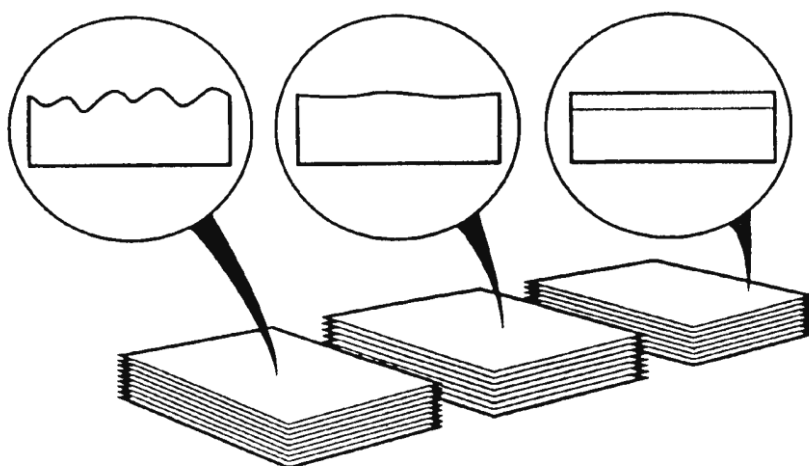


Рис. 6.3. Рельєф поверхні паперу

На сьогодні широко застосовуються шість технологій для кольорового друку. Вони реалізуються в ударних («голкових») матричних принтерах (*dot matrix*), у струменевих принтерах із рідким чорнилом (*liquid ink-jet*), у принтерах із термопереносом воскової мастики (*thermal wax transfer*), у принтерах із термосублімацією фарбника (*dye sublimation*), у струменевих принтерах зі зміною фази фарбника (*phase-change ink-jet*) і в кольорових лазерних принтерах (*colour laser*).

Матричні принтери

Як відомо, ідея матричних друкуючих пристроїв полягає в тому, що необхідне зображення відтворюється з набору окремих точок, що наносяться на папір тим або іншим способом. Нагадаємо також, що практично всі друкуючі пристрої

(за винятком, мабуть, сторінкових) можуть бути ударними (*impact*) і ненаголошеними (*non-impact*). Принцип роботи кольорових ударних матричних принтерів полягає в тому, що вертикальний ряд (або два ряди) голок «вбиває» фарбник із стрічки прямо в папір. На відміну від звичайних монохромних пристроїв, в останньому випадку використовується багатоколірна стрічка. Система управління цих принтерів піклується не тільки про конкретну голку, але й про кольорові стрічки. Відразу відзначимо, що крім шуму, властивого всім ударним пристроям, швидкість, палітра і якість кольорів у даному випадку, як правило, незадовільні. Це, утім, стосується не тільки паперу, але й плівок. Відмітимо також, що з часом відтворені кольори стають тьмянішими, оскільки зі збільшенням терміну служби стрічка забруднюється. Це пов'язано в основному з прямим контактом багатоколірної стрічки з кольоровим зображенням, що виводиться. До достоїнств подібних пристроїв можна віднести надійність, низьку вартість сторінки зображення, можливість друку на звичайному папері. Ударні кольорові матричні принтери в основному знаходять застосування при виведенні нескладних зображень. Ціна таких пристроїв відносно невисока – близько 800 дол.

Струменева технологія друку (*Liquid ink-jet*) є на сьогодні найпоширенішою для кольорових пристроїв. Струменеві чорнильні принтери підрозділяються на пристрої безперервної (*continuous drop, continuous jet*) і дискретної (*drop-on-demand*) дії. Останні знову ж таки діляться на дві категорії: із нагріванням чорнила («бульбашкова» технологія – *bubble-jet*, або *thermal ink-jet*) і засновані на дії п'єзо-ефекту (*piezo*).

У простому випадку принцип дії пристрою за технологією *continuous jet* заснований на тому, що струмінь чорнила, що постійно випускається з сопла друкуючої голівки, прямує або на папір (для нанесення зображення), або на спеціальний приймач, звідки чорнила знову потрапляють у загальний резервуар. У робочу камеру чорнило подається мікронасосом, а елементом, що задає їх рух, є, як правило, п'єзодатчик. Описаний вище принцип дії друкуючого пристрою використовує сьогодні дуже невелика кількість принтерів. Виробництвом кольорових принтерів, що використовують дану технологію, займається, наприклад, фірма Iris Graphics.

При реалізації *bubble-jet*-методу в кожному соплі друкуючої голівки знаходиться маленький нагрівальний елемент (наприклад, тонкоплівний резистор). При пропусканні струму через тонкоплівний резистор останній за декілька мікросекунд нагрівається до температури близько 500 градусів і віддає тепло,

що виділяється, безпосередньо оточуючому його чорнилу. При різкому нагріванні утворюється чорнильний паровий міхур, який прагне виштовхнути через вихідний отвір сопла краплю рідкого чорнила. Оскільки при відключенні струму тонкоплівний резистор також швидко остигає, паровий міхур, зменшуючись у розмірах, «підсмоктує» через вхідний отвір сопла нову порцію чорнила, яке займає місце «вистреленої» краплі. Кольорові принтери від фірм Canon і Hewlett-Packard використовують саме цю технологію.

Як вже було сказано, другий метод для управління соплом заснований на дії діафрагми, сполученої з п'єзоелектричним елементом. Як відомо, зворотний п'єзоефект полягає в деформації п'єзокристала під впливом електричного поля. Зміна розмірів п'єзоелементу, розташованого збоку вихідного отвору сопла і пов'язаного з діафрагмою, приводить до викидання краплі і приливу через вхідний отвір нової порції чорнила. Подібні пристрої випускаються компаніями Epson, Brother, Data-products і Tektronix. До речі, фірмою Epson запропонований новий тип багатошарової п'єзоелектричної голівки, яка усуває «сателіти» – маленькі крапельки, супроводжуючі основну краплю. Чіткість у цьому випадку підвищується в основному для монохромних зображень.

Відмітимо, що сопла (канальні отвори) на друкуючій голівці струменевих принтерів, через які розбризкується чорнило, відповідають «ударним» голкам матричних принтерів. Оскільки розмір кожного сопла істотно менший за діаметр голки (тонше за людське волосся), а кількість сопел може бути більшою, то одержуване зображення теоретично повинне бути в цьому випадку чіткіше. На жаль, це не завжди так, і дуже багато чого залежить від якості використовуваного паперу. Справа в тому, що чорнило має властивості просочуватися (куди не треба), розтікатися і змішуватися до висихання. Це призводить до зниження яскравості, а також до зміни кольоровості зображення.

Для того, щоб подолати всі ці неприємності, використовуються різні підходи. Наприклад, хіміки фірми DuPont розробили для принтерів компанії Hewlett-Packard спеціальне пігментне чорнило (правда, теж не без недоліків). А ось щоб уникнути змішування чорнила, у моделі принтера IBM Color JetPrinter PS4079 фірми Lexmark передбачені паузи між проходами для нанесення первинних кольорів. Згадувана трохи вище компанія Hewlett-Packard для тієї ж мети (висихання чорнила) використовує підігрів носія, тобто паперу. Такий метод боротьби із змішуванням чорнила реалізований у моделях HP PaintJet XL300 і DeskJet 1200C.

Отже, до основних достоїнств технології *continuous jet* відноситься можливість відтворення широкої палітри кольорів з високою якістю, проте при невисокій швидкості друку вартість подібних кольорових принтерів досягає декількох десятків тисяч доларів.

Пристрої дискретної дії (*drop-on-demand*) достатньо дешеві (від 500 дол. і вище) і також дозволяють одержувати широку гамму кольорів, проте вимагають, як правило, спеціального паперу.

Принцип роботи принтера з термопереносом (*Thermal wax transfer*) полягає в тому, що термопластична фарбувальна речовина, нанесена на тонкій підкладці, потрапляє на папір саме в тому місці, де нагрівальними елементами (аналогами сопел і голлок) друкуючої голівки забезпечується належна температура (близько 70–80 °C). Конструктивно такий спосіб друку достатньо простий, до того ж він забезпечує практично безшумну роботу. Для нанесення кольорового зображення потрібно три або чотири проходи: по одному для первинних кольорів і ще один у разі використання окремого чорного кольору, що, відповідно, збільшує час друку. Принтери, які використовують дану технологію, звичайно вимагають спеціального паперу. Вартість виведеної сторінки із зображенням, як правило, вища, ніж для струменевих принтерів. Для даних пристроїв також характерна невелика швидкість друку (1–2 сторінки за хвилину). Проте принтери з термопереносом – достатньо надійні пристрої, які не вимагають складного обслуговування і можуть відтворювати кольорове зображення (до 16,7 млн кольорів) як на плівці, так і на папері з роздільною здатністю 200–300 dpi (точок на дюйм). Вартість подібних пристроїв може складати від 1 до 10 тис. дол.

Ще один клас кольорових друкуючих пристроїв – так звані принтери з термосублімацією (*Dye sublimation*). Ця технологія найбільш близька до технології термопереноса, тільки елементи друкуючої голівки нагріваються в даному випадку вже до температури близько 400 °C. Хоча, можливо, термін «термосублімація» не дуже вдалий, але він достатньо чітко пояснює, яким чином фарбувальній речовині передається необхідна порція енергії сублімації. Нагадаємо, що під сублімацією розуміють перехід речовини із твердого стану в газоподібний минаючи стадію рідини (наприклад, кристали йоду сублімують при нагріванні). Таким чином, порція фарбника сублімується з підкладки і осідає на папері або іншому носії. У принтерах з термосублімацією фарбника є можливість точного визначення необхідної кількості фарбника, що переноситься на папір (наприклад, 19 % cyan, 65 % magenta, 34 % yellow). Комбінацією кольорів фарбників можна підібрати практично будь-яку колірну палітру.

Дана технологія використовується тільки для кольорового друку, а реалізуючи її пристрої звичайно відносяться до класу *high end*. До їх основних переваг належать практично фотографічна якість одержуваного зображення і широка гамма відтінків кольорів без використання растрівання. Основним обмеженням застосування даних принтерів є висока вартість кожної копії зображення (більше долара за сторінку).

Принтери, що використовують технологію *Phase change ink-jet*, називаються також принтерами з твердим фарбником. Принцип роботи таких пристроїв приблизно наступний. Воскові стрижні для кожного первинного кольору фарбника поступово розплавляються спеціальним нагрівальним елементом при температурі близько 90 градусів і потрапляють в окремі резервуари.

Розплавлені фарбники подаються звідти спеціальним насосом у друкуючу голівку, що працює звичайно на основі п'єзоефекту. Краплі воскоподібного фарбника на папері застигають практично миттєво, але забезпечують необхідне з нею зчеплення. На відміну від звичайної технології *liquid ink-jet*, у даному випадку не відбувається ні просочування, ні розтікання, ні змішення фарбників. Саме тому принтери, що використовують технологію *phase change ink-jet*, працюють із будь-яким папером. Якість кольорів виходить просто чудова, до того ж допустимий і двосторонній друк. Вартість однієї копії невисока, як, утім, і швидкість друку (близько 2 сторінок за хвилину).

У лазерних принтерах (*Colour laser*) використовується електрографічний принцип створення зображення – приблизно такий же, як і в копіювальних машинах. Найбільш важливими частинами лазерного принтера можна вважати фотопровідний барабан (або стрічку), напівпровідниковий лазер і прецизійну оптико-механічну систему, що переміщає промінь. Лазер формує електронне зображення на фоточутливій стрічці послідовно для кожного кольору тонера (СМΥК). Тобто принтер, що працює в монохромному режимі із швидкістю 8 сторінок/хв, у кольоровому режимі забезпечить тільки 2 сторінки/хв. Коли зображення на фоточутливій стрічці повністю побудоване, тонер з барабана притягується до паперу. Після цього зображення закріплюється на ньому за рахунок нагріву частинок тонера до температури плавлення. Остаточну фіксацію зображення здійснюють спеціальні вали, що притискають розплавлений тонер до паперу.

Технологічно даний процес здійснюється досить непросто, тому ціни на кольорові лазерні принтери до недавнього часу складали декілька десятків тисяч доларів.

Деякі характеристики ряду моделей кольорових принтерів, що випускаються фірмою Tektronix, наведені в таблиці 6.2.

Таблиця 6.2

Характеристики кольорових принтерів

Модель принтера	Phaser 200E	Phaser 200i	Phaser 111PX	Phaser 11sd
Метод друку	Thermal wax	Thermal wax	Solid inc-jet	dye sublimation
Формат паперу	A4	A4	A3	A4
Максимальна швидкість друку, сторінок/хв монохромний кольоровий	4 2	4 2	1 0,5	1 0,1
Максимальна роздільна здатність, dpi	300×300	300×300	300×300	300×300
Об'єм пам'яті, Мб	4	6	10	16
Інтерфейси	Centronics RS-232C	Centronics RS-232C EtherTalk	Centronics RS-232C EtherTalk	Centronics RS-232C EtherTalk
Кількість резидентних шрифтів	17	39	39	39
Розміри, мм	279×394× 444	279×394× 444	н/д	363×432× 686
Вага, кг	18,2	18,2	н/д	33,1

6.5. ПРИСТРОЇ ВВЕДЕННЯ ДАНИХ

Як відомо, клавіатура є стандартним пристроєм введення тексту. Тому, якщо ви багато працюєте з текстом, то вам варто обрати зручну клавіатуру. Що стосується операційної системи, ігор і багатьох програм, то переважна більшість користувачів вважають за краще працювати з ними за допомогою мишки. Для деяких, можливо, буде відкриттям, що майже все це робиться і за допомогою клавіатури, причому іноді значно швидше (якщо використовувати комбінації «гарячих клавіш»). Такий підхід не набув поширення тому, що доводиться тримати в пам'яті всі ці комбінації.

З погляду механізму, використовуваного при роботі клавіш, клавіатури підрозділяються на такі основні категорії: механічні, напівмеханічні, мембранні і герконові.

Механічна клавіатура

На платі такої клавіатури під кожною клавішею знаходиться пара контактів з цілісного металу. При натисненні на клавішу

вони замикаються, а розмикаються за допомогою металевої пружинки. Такі клавіатури досить дорогі, але при цьому дуже надійні і довговічні. Правда, є в них ще один маленький недолік: при попаданні рідини і маленьких твердих предметів (у вигляді крихт і т. ін.) клавіатура впадає в «коматозний» стан або зовсім «відключається».

Мембранна клавіатура

Тут все простіше: у наявності дві мембрани: одна – на платі, інша – під клавішею. Як і у випадку з механічною клавіатурою, відбувається замикання контактів, а розмикається це все за допомогою невеликого гумового куполу. Особливість такої конструкції полягає в тому, що контакти мембран знаходяться усередині куполу, так що гарантована повна герметичність і 100-відсотковий захист. Основні плюси даних клавіатур – низька ціна, малий шум, що видається під час набору тексту. Правда, мембрани не настільки міцні, як металеві контакти.

Напівмеханічна клавіатура

Це гібрид першого і другого різновидів. Від мембранної клавіатури напівмеханічній клавіатурі дісталася герметичність, а з нею всі переваги і всі недоліки (захист від вологи і недовговічність захисного механізму клавіші). А від механічної – надійніші контакти, що менш стираються. При цьому контакти розмикаються пружиною, а повернення клавіші здійснюється гумовими куполами. За довговічністю та вартістю така клавіатура є чимось середнім порівняно з раніше описаними різновидами.

Герконова клавіатура

Для того, щоб розібратися з механізмом роботи, потрібно вивчити основи теорії електромагнітного поля, а якщо коротко: у ній є геркони і випромінювачі. Геркони – це пластинки (вище іменовані контактами), розташовані в герметичній конструкції і чутливі до магнітної дії. Далі – випромінювач (магніт). Від зовнішньої дії клавіатура зовсім не захищена. Проте в порівнянні з її високою вартістю, це не такий істотний мінус. До плюсів, відповідно, можна віднести «вічність». При натисненні на клавішу виразно чутний її клік (перемикання геркона). Такі клавіатури раніше були досить поширені, але тепер уже практично не зустрічаються.

Крім типу необхідно враховувати ще деякі характеристики клавіатур. Наприклад, наявність підставки для рук. Навіщо? Справа в тому, що в людей, що сидять за комп'ютером і набирають велику кількість текстів, досить часто сильно болять руки. Це так званий «тунельний синдром». При дружі ваші руки практично не рухаються, а, спираючись зап'ястями на край столу, ви пережимаєте судини, що живлять долоню і пальці, і тим самим травмуєте нерв. Якщо ж зап'ястя рук розташувати

на спеціальній підставці, ситуація зміниться в кращу сторону, і ви зможете працювати довше і продуктивніше.

Також до важливих параметрів клавіатури можна віднести хід клавіш, ергономічність, розкладку.

Тепер від технічних характеристик до топографії. Буквено-цифрові клавіші – це основна частина клавіатури, за допомогою якої вводиться текст. Латинські букви і розділові знаки розташовані завжди однаково, а ось форма і розміри важливих службових клавіш можуть істотно відрізнятися. Найчастіше відрізняються правий Shift (тимчасова зміна регістра), Backspace (видалення однієї помилково набраної букви) і Enter (введення). Тут же розташована одна символна клавіша (зворотний слеш «\»), яка використовується в основному при вказівці імен файлів і каталогів. Нерідко зустрічається Enter висотою в два середні ряди клавіш, формою що нагадує перевернену букву «Г». Над ним розташований Backspace, під ним – Shift. Для зручності користувача всі ці кнопки бажано зробити довгими, але на практиці здійснити це складно – не дозволяють пропорції клавіатури. Тому іноді укорочують Backspace (більш-менш прийнятний варіант) або Shift (це гірше), зрідка модифікують форму Enter. Останній варіант найбільш прийнятний, бо навіть зменшений Enter залишається достатньо великою і зручною кнопкою.

Слід згадати і про клавіші нижнього ряду, а саме: пропуску, Ctrl, Alt і т. ін. Дуже зручний подовжений Ctrl, але важливо, щоб це розширення не дуже зменшило площу такої важливої клавіші, як пропуск. Він повинен залишатися довшим, ніж всі разом узяті кнопки, розташовані зліва від нього.

Важливо, щоб кириличні букви помітно відрізнялися від латинських. Звичайно латиницю наносять сірим кольором, а букви кирилиці – червоним. Хорошим тоном вважається робити букви цих двох розкладок трохи різними за розміром. Розташування букв стандартизовано, але зустрічаються проблеми з розділовими знаками. На деяких клавіатурах використовується стара розкладка, де крапка і кома розташовані на цифрах «6» і «7», тоді як у Windows вони прив'язані до клавіші «/».

Навігаційні клавіші – цей блок знаходиться праворуч від алфавітно-цифрового і складається (у класичному варіанті) з трьох груп клавіш. Внизу розташовані 4 кнопки із стрілками, вони мають форму переверненої «Т». Вище – шість клавіш для навігації і правки текстів: Home (повернення до початку), PgUp (на одну сторінку вгору) і т. ін. Вони використовуються рідше. Ще вище – кнопки Pause, Scroll Lock і Print Screen, які можна назвати функціональними. Форма і розташування клавіш такі,

що вони зручні як в іграх, так і в робочих додатках, будь-яку кнопку легко знайти усліпу. До них швидко звикаєш, тому будь-які відхилення від стандартної розкладки в цій зоні вимагають серйозного підходу. Трапляється, що такі нововведення виявляються незручними при роботі.

Також у правій частині знаходиться блок клавіш із цифрами (зручно використовувати, якщо більша частина роботи ведеться з числами).

Функціональні клавіші – це ряд над алфавітно-цифровим блоком, що складається з кнопок Escape, F1–F12. Відмінності тут в основному полягають у висоті клавіш, зрідка міняють їх компонування (групи по чотири або по шість клавіш).

6.6. ПРИСТРОЇ ЗБЕРІГАННЯ ДАНИХ

Стримери

Стримери – це накопичувачі на магнітній стрічці. Їх відрізняє порівняно низька ціна. До недоліків стримерів відносять малу продуктивність (вона пов'язана перш за все з тим, що магнітна смужка – це пристрій послідовного доступу) і недостатню надійність (окрім електромагнітних наведень стрічки стримерів піддані підвищеним механічним навантаженням і можуть фізично виходити з ладу).

Місткість магнітних касет (картриджів) для стримерів досягає декількох десятків гігабайт. Подальше підвищення місткості за рахунок підвищення щільності запису знижує надійність зберігання, а підвищення місткості за рахунок збільшення довжини стрічки стримується низьким часом доступу до даних.

Накопичувачі на зйомних магнітних дисках

До цієї категорії відноситься декілька різних типів пристроїв, жоден з яких, так і не став загальноприйнятим стандартом. Наприклад, ZIP-накопичувачі випускаються компанією Lomega, що спеціалізується на створенні зовнішніх пристроїв для зберігання даних. Пристрій працює з дисковими носіями, що за розміром трохи перевищують стандартні гнучкі диски і мають місткість 100/250/750 Мб. Основним недоліком ZIP-накопичувачів є несумісність їх зі стандартними гнучкими дисками 3,5 дюйма. Такою сумісністю володіють пристрої HiFD компанії SONY. Вони дозволяють використовувати як спеціальні носії місткістю 200 Мб, так і звичайні гнучкі диски. Розповсюдження цих пристроїв стримується високою ціною.

Накопичувачі JAZ, як і ZIP-накопичувачі, випускаються компанією Lomega. За своїми характеристиками JAZ-носій наближається до жорстких дисків, але на відміну від них є змінним. Залежно від моделі накопичувача на одному диску можна розмістити 1 або 2 Гб даних.

Магнітооптичні пристрої

Ці пристрої одержали широке застосування в комп'ютерних системах високого рівня завдяки своїй універсальності. З їх допомогою розв'язуються завдання резервного копіювання обміну даними і їх накопичення. Проте достатньо висока вартість приводів і носіїв не дозволяє віднести їх до пристроїв масового вжитку.

Флеш-диски

Це сучасний пристрій зберігання даних на основі незалежної флеш-пам'яті. Пристрій має мінімальні розміри і допускає «гаряче» підключення в роз'єм USB, після чого розпізнається як жорсткий диск, причому не вимагає установки драйвера.

6.7. ЗАСОБИ МУЛЬТИМЕДІА

Мультимедіа – це інтерактивні системи, що забезпечують роботу з нерухомими зображеннями і рухомих відео, анімованою комп'ютерною графікою і текстом, мовою і високоякісним звуком.

Поява систем мультимедіа, безумовно, проводить революційні зміни в таких галузях, як освіта, комп'ютерний тренінг, у багатьох сферах професійної діяльності, науки, мистецтва, у комп'ютерних іграх і т. ін.

Поява систем мультимедіа обумовлена як вимогами практики, так і розвитком теорії. Проте різкий ривок, що відбувся в цьому напрямі за останні декілька років, забезпечений перш за все розвитком технічних і системних засобів. Це і прогрес у розвитку ПК: різко збільшені об'єм пам'яті, швидкодія, графічні можливості, характеристики зовнішньої пам'яті, і досягнення в галузі відеотехніки, лазерних дисків – аналогових і CD-ROM, а також їх масове впровадження. Важливу роль зіграло і розроблення методів швидкого і ефективного стиснення/розгортки даних.

Сучасний мультимедіа-ПК у повному «озброєнні» нагадує домашній стереофонічний Hi-Fi комплекс, об'єднаний з дисплеєм-телевізором. Він укомплектований активними стереофонічними колонками, мікрофоном і дисководом для оптичних компакт-дисків CD-ROM (CD – *Compact Disc*, компакт-диск; ROM – *Read only Memory*, пам'ять тільки для читання). Крім того, у середині комп'ютера розміщено новий для ПК пристрій – аудіоадаптер, що дозволив перейти до прослуховування чистих стереофонічних звуків через акустичні колонки з вбудованими підсилювачами.

Розглянемо деякі технічні питання, що стосуються мультимедіа. Основна проблема – спільне оброблення різнорідних

даних: цифрових і аналогових, «живого» відео і нерухомих зображень і т. ін. У комп'ютері всі дані зберігаються в цифровій формі, тоді як теле-, відео- і більшість аудіоапаратури має справу з аналоговим сигналом. Вихідні пристрої комп'ютера – монітори і динаміки – також мають аналоговий вихід. Тому простий і найбільш дешевий шлях побудови перших систем мультимедіа полягав у стиковці різноманітної апаратури з комп'ютером, наданні комп'ютеру можливостей управління цими пристроями, поєднання вихідних сигналів комп'ютера та відео- і аудіопристроїв і забезпеченні їх нормальної спільної роботи. Подальший розвиток мультимедіа відбувається у напрямі об'єднання різноманітних типів даних у цифровій формі на одному середовищі-носії, у рамках однієї системи.

6.7.1. Відео і аудіо засоби

При поєднанні сигналів основні проблеми виникають з відео-зображенням. Різні ТВ-стандарти, що існують у світі (NTSC, PAL, SECAM), застосування різних моніторів і відеоконтролерів спричиняють різноманітність підходів у вирішенні виникаючих проблем. Проте у будь-якому випадку потрібна синхронізація двох зображень, для чого слугує пристрій генлок (*genlock*). З його допомогою на екрані монітора можуть бути суміщені зображення, згенеровані комп'ютером (анімована або нерухома графіка, текст, титри), і «живе» відео. Якщо додати ще один пристрій – кодер (*encoder*), комп'ютерне зображення може бути перетворене у форму ТВ-сигналу і записане на відеоплівку. «Настільні відео-студії», застосування систем мультимедіа, що є одним із прикладів, дозволяють готувати суміщені відео-комп'ютерні кліпи, титри для відеофільмів, допомагають при монтажі кінофільмів.

Системи такого роду не дозволяють якимось обробляти або редагувати саме аналогове зображення. Щоб це стало можливим, його необхідно оцифрувати і ввести в пам'ять комп'ютера. Для цього призначена так звана плата захоплення (*capture board, frame grabbers*). Оцифрування аналогових сигналів породжує величезні масиви даних. Так, кадр стандарту NTSC (525 рядків), перетворений платою типу Truevision, перетворюється на комп'ютерне зображення з роздільною здатністю 512×482 пікселів. Якщо кожна точка представлена 8 бітами, то для зберігання всієї картини потрібно близько 250 Кб пам'яті, причому погіршується якість зображення, оскільки забезпечується тільки 256 різних кольорів. Уважається, що для адекватного передання початкового зображення потрібно 16 млн відтінків, тому використовується 24-бітовий формат зберігання кольорової

картинки, а необхідний розмір пам'яті зростає. Оцифрований кадр може потім бути змінений, відредагований звичайним графічним редактором, можуть бути прибрані або додані деталі, змінені кольори, масштаби, додані спецефекти типу мозаїки, інверсії і т. ін. Природно, інтерактивне екранне оброблення можливе лише в межах роздільної здатності, що забезпечується даним конкретним відеоадаптером. Оброблені кадри можуть бути записані на диск у будь-якому графічному форматі і потім використовуватися як реалістичний нерухомий фон для комп'ютерної анімації. Можливе також покадрове оброблення початкового зображення і вивід назад на відеоплівку для створення псевдореалістичного мультфільму.

Запис послідовності кадрів у цифровому вигляді вимагає від комп'ютера великих об'ємів зовнішньої пам'яті: частота кадрів в американському ТВ – стандартні NTSC – 30 кадрів/с (PAL, SECAM – 25 кадрів/с), так що для запам'ятовування однієї секунди повнокольорового повноекранного відео потрібно 20–30 Мб, а оптичний диск місткістю 600 Мб вміщатиме менш півхвилини зображення. Але послідовність кадрів недостатньо тільки запам'ятати, її треба ще вивести на екран у відповідному темпі. Подібною швидкістю передання інформації – близько 30 Мб/с – не володіє жоден з існуючих зовнішніх запам'ятовуючих пристроїв. Щоб виводити на екран комп'ютера оцифроване відео, доводиться йти на зменшення об'єму передаваних даних (виведення зменшеного зображення в невеликому вікні, зниження частоти кадрової розгортки до 10–15 кадрів/с, зменшення числа біт/пікселів), що, у свою чергу, приводить до погіршення якості зображення.

Радикальніше обидві проблеми – пам'яті і пропускну здатності – розв'язуються за допомогою методів стиснення/розгортки даних, які дозволяють стискати інформацію перед записом на зовнішній пристрій, а потім прочитувати і розгортати в реальному режимі часу при виведенні на екран. Так, для рухомих відео-зображень існуючі адаптивні різницеві алгоритми можуть стискати дані з коефіцієнтом порядку 100 : 1–160 : 1, що дозволяє розмістити на CD-ROM близько години повноцінного озвученого відео. Робота цих алгоритмів заснована на тому, що звичайно подальший кадр відрізняється від попереднього лише деякими деталями, тому, узявши будь-який кадр за базовий, для наступних можна зберігати тільки відносні зміни. При значних змінах кадру, наприклад, при монтажному склеюванні, наїзді або панорамуванні камери, автоматично вибирається новий базовий кадр. Для статичних зображень коефіцієнт стиснення, природно, нижче – близько 20–30 : 1. Для аудіоданих застосовують свої методи компресії.

Існує симетрична і асиметрична схеми стиснення даних. При асиметричній схемі інформація стискається у автономному режимі (тобто одна секунда початкового відео стискається протягом декількох секунд або навіть хвилин могутніми паралельними комп'ютерами і поміщається на зовнішній носій, наприклад CD-ROM. На машинах користувачів встановлюється порівняно дешева плата декодування, що забезпечує відтворення інформації мультимедіа в реальному часі. Використання такої схеми збільшує коефіцієнт стиснення, покращує якість зображення, проте користувач позбавлений можливості розробляти власні продукти мультимедіа. При симетричній схемі стиснення і розгортка відбуваються в реальному часі на машині користувача, завдяки чому за персональними комп'ютерами і в цьому випадку зберігається їх основоположна цінність: з їх допомогою будь-який користувач має можливість здійснювати власну продукцію, у тому числі й комерційну, не виходячи з будинку. Правда при симетричній схемі дещо падає якість зображення: з'являються «змащені» кольори, картинка як би расфокусовується. З розвитком технології ця проблема поступово відходить, проте поки іноді віддають перевагу змішаній схемі, при якій розробник продукту створює, відладжує і випробовує продукт мультимедіа на своїй машині з симетричною схемою, а потім «напівфабрикат» у стандартному форматі відсилає на фірму, де його піддають стисненню на могутньому комп'ютері, з використанням більш довершених алгоритмів і розміщують продукт на CD-ROM.

У даний час цілий ряд фірм активно веде розроблення алгоритмів стиснення відеоінформації, прагнучи досягти коефіцієнта стиснення 200 : 1 і вище. В основі найбільш ефективних алгоритмів лежать різні адаптивні варіанти: DCT (*Discrete Cosine Transform*, дискретне косінус-перетворення), DPCM (*Differential Pulse Code Modulation*, різницева імпульсно-кодова модуляція), а також фрактальні методи. Алгоритми реалізуються апаратно – у вигляді спеціальних мікросхем, або firmware – записаної в ПЗП програми, або програмно.

Різницеві алгоритми стиснення застосовуються не тільки до відео-зображення, але й до комп'ютерної графіки, що дає можливість застосовувати на звичайних персональних комп'ютерах новий для них вид анімації, а саме: покадровий запис мальованих мультфільмів великої тривалості. Ці мультфільми можуть зберігатися на диску, а при відтворенні прочитуватися, розпаковуватися і видаватися на екран у реальному часі, забезпечуючи ті ж необхідні для плавного зображення 25–30 кадрів за секунду.

При використанні спеціальних відеоадаптерів (відеобластерів) мультимедіа-ПК стають центром побутової відеосистеми, що конкурує з самим найдосконалішим телевізором.

Новітні відеоадаптери мають засоби зв'язку з джерелами телевізійних сигналів і вбудовані системи захоплення кадру (компресії/декомпресії відеосигналів) у реальному часі, тобто практично миттєво. Відеоадаптери мають швидку відеопам'ять від 2 до 4 Мб і власні графічні процесори. Це дозволяє одержувати до 30–50 кадрів за секунду і забезпечити виведення рухомих повноекранних зображень.

Будь-який мультимедіа-ПК має у своєму складі плату-аудіоадаптер. Ці пристрої часто іменуються «саундбластерами» завдяки фірмі Creative Labs (Сінгапур), що назвала свої перші аудіоадаптери Sound Blaster. Аудіоадаптер дав комп'ютеру не тільки стереофонічне звучання, але й можливість запису на зовнішні носії звукових сигналів. Як уже було сказано раніше, дискові накопичувачі ПК зовсім не підходять для запису звичайних (аналогових) звукових сигналів, оскільки розраховані для запису тільки цифрових сигналів, які практично не спотворюються при їх переданні по лініях зв'язку.

Аудіоадаптер має аналого-цифровий перетворювач (АЦП), що періодично визначає рівень звукового сигналу і перетворює цей відлік на цифровий код. Він і записується на зовнішній носій уже як цифровий сигнал.

Цифрові вибірки реального звукового сигналу зберігаються в пам'яті комп'ютера (наприклад, у вигляді WAV-файлів). Зчитаний із диска цифровий сигнал подається на цифро-аналоговий перетворювач (ЦАП), який перетворює цифрові сигнали на аналогові. Після фільтрації їх можна підсилити і подати на акустичні колонки для відтворення. Важливими параметрами аудіоадаптера є частота квантування звукових сигналів і розрядність квантування.

Частоти квантування показують, скільки разів за секунду беруться вибірки сигналу для перетворення на цифровий код. Звичайно вони лежать у межах від 4–5 КГц до 45–48 КГц. У таблиці 6.3 показана залежність частоти квантування від виду сигналу.

Розрядність квантування характеризує число ступенів квантування і змінюється ступенем числа 2. Так, 8-розрядні аудіоадаптери мають $2^8 = 256$ ступенів, що явно недостатньо для високоякісного кодування звукових сигналів. Тому зараз застосовуються в основному 16-розрядні аудіоадаптери, що мають $2^{16} = 65536$ ступенів квантування – як у звукового компакт-диска.

Залежність частоти квантування від виду сигналу

Частотний діапазон	Вид сигналу	Частота квантування
400–3500 Гц	Мова (ледве розбірлива)	5,5 КГц
250–5500 Гц	Мова (середня якість)	11,025 КГц
40–10000 Гц	Якість звучання приймача ультракороткохвильового сигналу	22,040 КГц
20–20000 Гц	Звук високої якості	44,100 КГц

Інший спосіб відтворення звуку полягає в його синтезі. Під час вступу на синтезатор деякої управляючої інформації за нею формується відповідний вихідний сигнал. Сучасні аудіоадаптери синтезують музичні звуки двома способами: методом частотної модуляції FM (*Frequency Modulation*) і за допомогою хвильового синтезу (вибираючи звуки з таблиці звуків – *Wave Table*). Другий спосіб забезпечує звучання, близьке до натурального.

Частотний синтез (FM) з'явився в 1974 р. (PC-Speaker). У 1985 році з'явився AdLib, який, використовуючи частотну модуляцію, був здатний грати музику. Нова звукова карта Sound Blaster уже могла записувати і відтворювати звук. Стандартний FM-синтез має середні звукові характеристики, тому на картах встановлюються складні системи фільтрів проти можливих звукових перешкод.

Суть технології WT-синтезу полягає в наступному. На самій звуковій карті встановлюється модуль ПЗП з «защитими» в нього зразками звучання справжніх музичних інструментів – семплами, а WT-процесор за допомогою спеціальних алгоритмів навіть по одному тону інструменту відтворює решту звуків. Крім того, багато виробників оснащують свої звукові карти модуляторами ОЗП, так що є можливість не тільки записувати довільні семпли, але й підвантажувати нові інструменти.

До речі, управляючі команди, для синтезу звуку можуть надходити на звукову карту не тільки від комп'ютера, але й від іншого, наприклад, MIDI (*Musical Instruments Digital Interface*), пристрою. Власне, MIDI визначає протокол передання команд по стандартному інтерфейсу. MIDI-повідомлення містить посилення на ноти, а не запис музики як такий. Зокрема, коли звукова карта одержує подібне повідомлення, воно розшифровується (які ноти яких інструментів повинні звучати) і відпрацьовується на синтезаторі. У свою чергу, комп'ютер може через MIDI управляти різними «інтелектуальними» музичними інструментами з відповідним інтерфейсом.

Для електронних синтезаторів звичайно вказується кількість одночасно звучних інструментів і їх загальна кількість (від 20 до 32). Також важлива і програмна сумісність аудіоадаптера з типовими звуковими платформами (Sound Blaster, Roland, AdLib, Microsoft Sound System, Gravis Ultrasound та ін.).

Як приклад розглянемо склад вузлів одного з могутніх аудіоадаптерів – Sound Blaster AWE 32 Value. Він містить два мікрофонні малощумні підсилювачі з автоматичним регулюванням посилення для сигналів, що надходять від мікрофону, два лінійні підсилювачі для сигналів, що надходять із лінії, з програвача звукових дисків або музичного синтезатора. Крім того, сюди входять програмно управляючий електронний мікшер, що забезпечує поєднання сигналів від різних джерел і регулювання їх рівня і стереобалансу, 20-голосий синтезатор музичних звуків частотної модуляції FM, програмно керований хвильовий (табличний) синтезатор музичних звуків і звукових ефектів (16 каналів, 32 голоси, 128 інструментів), аналого-цифровий 16-розрядний перетворювач для перетворення аналогового сигналу з виходу мікшера в цифровий сигнал, систему стиснення цифрової інформації з можливістю застосування розширеного звукового процесора ASP. Нарешті, аудіоадаптер має цифро-аналоговий перетворювач (ЦАП) для перетворення цифрових сигналів, що несуть інформацію про звук, на аналоговий сигнал, адаптивний електронний фільтр на виході ЦАП, що знижує перешкоди від квантування сигналу, двохканальний підсилювач потужності по 4 Вт на канал з ручним і програмно керованим регулятором гучності і MIDI-роз'єм для підключення музичних інструментів.

Як видно з цього переліку, аудіоадаптер – досить складний технічний пристрій, побудований на основі використання останніх досягнень в аналоговій і цифровій аудіотехніці.

У новітні звукові карти входить цифровий сигнальний процесор DSP (*Digital Signal Processor*) або розширений сигнальний процесор ASP (*Advanced Signal Processor*). Вони використовують довершені алгоритми для цифрової компресії і декомпресії звукових сигналів, для розширення бази стереозвуку, створення луни і забезпечення об'ємного (квадрофонічного) звучання. Програма підтримки ASP QSound поставляється безкоштовно фірмою Intel на CD-ROM «Software Developer CD». Важливо відзначити, що процесор ASP використовується при звичайних двохканальних стереофонічних запису і відтворенні звуку. Його застосування не завантажує акустичні тракти мультимедіа комп'ютерів.

6.7.2. Носії інформації

Важливою проблемою мультимедіа є забезпечення адекватних засобів доставки, розповсюдження мультимедійної – інформації. Носії повинні вміщати величезні об'єми різнорідної інформації, дозволяти швидкий доступ до окремих її компонентів, якісне їх відтворення, і при цьому бути досить дешевими, компактними і надійними. Ця проблема була вирішена лише з появою оптичних дисків різних типів.

Формати сучасних носіїв інформації відображені в таблиці 6.4.

Таблиця 6.4

Формати носіїв інформації

Формат	Опис
CD-Audio	Старий формат компакт-дисків. Майже всі дисководи CD-ROM можуть програвати звукові компакт-диски
CD-Interactive	Власний формат Philips для «інтерактивних», в основному ігрових, компакт-дисків для домашніх програвачів
CD-ROM/XA	Поеднує стислі дані і звук, а так само змішаний режим, записуються з чергуванням для рівнішого відтворення. Кращий формат для мультимедіа
Mixed mode	Комбінація звуку у форматі Red Book і даних CD-ROM. Перша доріжка повинна містити дані, за нею можуть слідувати доріжки CD-Audio
CD-Plus	Схожий із режимом Mixed mode, відмінність – запобігання зверненню звукового програвача до доріжок із даними, щоб уникнути пошкодження динаміків
ISO-9660	Стандартний формат і структура каталогів для CD-ROM
HFS (Hierarhical File Structure)	Формат даних, розроблений для Macintosh
Hybrid discs	Містить системи HFS і ISO
Photo CD	Розроблений фірмою Kodak для запису фотографій високої якості. Для відтворення необхідний пристрій CD-ROM/XA або CD-Interactive
Video CD	Відеоінформація у форматі MPEG-1 і звук. Стандарт призначений для відтворення фільмів

У перших системах мультимедіа були використані аналогові диски – їх звичайно називають «відеодисками». Діаметр цих дисків – 12 або 8 дюймів. Відомі 12-дюймові диски стандарту LV (*Laser Vision*), що підтримуються Sony, Philips і Pioneer.

Інформація записується на лазерний диск по спіралі, кожен виток цієї спіралі називається доріжкою. Існують 2 способи запису інформації на лазерні диски – CAV (*Constant Angular*

Velocity – з постійною кутовою швидкістю) і *CLV* (*Constant Linear Velocity* – з постійною лінійною швидкістю). При записі *CLV*-диски вміщують по 1 год відео на кожній зі сторін (диски *CLV* називають також «довгограючими»), проте їх інтерактивні можливості обмежені, тому вони в системах мультимедіа використовуються рідко, частіше застосовуються при записі фільмів.

Диск *CAV* вміщує на кожній доріжці один відеокадр (точніше, два напівкадри, що містять парні і непарні рядки кадру – телевізор працює в інтерлейсному режимі, поперемінно висвічуючи парні і непарні рядки кожного кадру). Диск обертається з постійною швидкістю 30 об./с, забезпечуючи необхідні для *NTSC* 30 кадрів/с. Кожна зі сторін диска має 54 тис. доріжок, тобто вміщує 30 хв відео *NTSC* (диски для *PAL* – 37 хв). Кожен кадр має свій номер, або адресу, за номером можливий прямий доступ до будь-якого кадру. Кадри можуть трактуватися як нерухомі зображення – для цього після завершення зчитування доріжки пристрій не переходить на наступну, а знов прочитує ту ж саму); можливе також програвання з різними швидкостями і у зворотному напрямі. Разом із зображенням записуються дві звукові доріжки, доступні, утім, тільки при перегляданні кадрів у режимі відео. Інформацію на диску можна розбити на частини – до 80 частин на кожній зі сторін. Управляюча інформація – номери кадрів, номери частин – поміщається в «бланкових» (невидимих) частинах кадрів.

Проміжний, «аналого-цифровий» формат лазерних дисків – *LVRM*, або *AIV* (*Advanced Interactive Video* – покращуване інтерактивне відео) – дозволяє поєднувати на одному диску аналогове відео з цифровим звуком і даними.

Нарешті існують різні типи цифрових дисків: *CD-ROM*, *WORM*, *CD-ROM* з можливістю перезапису, як і цифрові аудіо-компакт-диски *CD-DA* (*Compact Disc – Digital Audio*) мають діаметр 5,25 дюйма; вони вміщують 500–600 Мб інформації і є зараз найбільш масовим цифровим засобом доставки мультимедіа-інформації.

CD-ROM – кружок із прозорої пластмаси, полікарбонату, на одній із поверхонь якого нанесений тонкий світловідбиваючий шар. Цей сріблястий шар добре видно з тильної сторони прозорого диска. У ньому є мікроскопічні поглиблення – піти, створені в процесі його копіювання з оригіналу.

Типова довжина піти 0,8–3,2 мкм, ширина – 0,4 мкм, глибина – 0,12 мкм, а відстань між окремими доріжками 1,6 мкм. На одному дюймі (2,54 см) поверхні диска розміщується 16 тис. доріжок (для порівняння – на одному дюймі магнітного диска поміщається тільки 96 доріжок). Завдяки таким малим розмірам

птів звичайний CD-ROM вміщає величезний об'єм інформації – близько 700 Мб. Нові типи дисків мають на порядок більший об'єм і допускають запис інформації користувачем.

Робочою є тільки одна поверхня диска CD-ROM. Вона захищена товстим шаром лаку, на який звичайно наноситься барвиста етикетка. У програвачі диск обернений цією стороною назовні. Протилежна (тильна) сторона використовується для зчитування лазерним променем. Промінь проходить крізь неї, оскільки основа диска – прозора пластмаса. Товщина диска – 1,2 мм, зовнішній діаметр – 120 мм, діаметр внутрішнього отвору – 15 мм.

Швидкість обертання диска непостійна і змінюється від 500 об./хв для внутрішньої частини диска, з якою починається зчитування, до 200 об./хв для зовнішньої. Спеціальний оптико-електронний блок має пристрої для стабілізації випромінювання лазера, автоматичного фокусування, стеження за доріжкою при битті диска і вибору треків диска для зчитування.

Для зчитування інформації з CD-ROM використовується напівпровідниковий діод з фокусуючою і стежачою оптичною системою. Внутрішня поверхня диска, на яку кладуть диск на підставку (у касету) дисководу, знаходиться не у фокусі оптичної системи лазерного випромінювача. Діаметр світлової плями від лазера, що створює конус світла, сходиться близько 1 мм. Тому помірні забруднення неробочої поверхні, наприклад порошинки на ній, відбитки пальців і навіть невеликі подряпини, практично не впливають на відтворення. На відміну від звичних жорстких магнітних дисків, диски CD-ROM можна замінювати в лічені секунди. Адже один диск CD-ROM за місткістю рівний приблизно 500 звичайним гнучким дискам формату 3,5 дюймів на 1,44 Мб. Економія на дискетах є важливою характеристикою мультимедіа.

Програвачі комп'ютерних компакт-дисків, які звичайно називають CD-ROM-драйвами, бувають двох типів: зовнішні (зі своїм корпусом) і внутрішні – вбудовувані в системний блок комп'ютера. Останні нагадують накопичувачі на гнучких магнітних 5,25-дюймових дискетах і мають однакові з ним розміри.

На передній панелі дисководу CD-ROM звичайно є кнопка Eject для викиду або плавного висунення піддону, індикатор Busy (зайнято), роз'єм для підключення стереотелефонів і регулятор гучності, що використовується при програванні звукових дисків.

Повноцінне «озброєння» мультимедіа-ПК вимагає підключення до нього безлічі зовнішніх пристроїв: аудіо- і відеоадаптерів, телевізійних і радіо-тюнерів, дисководів CD-ROM,

джойстиків, клавіатури MIDI та ін. Усі вони обслуговуються масою програмних утиліт – драйверів – і нерідко конфліктують один з одним. У зв'язку з цим крупні розробники ПК об'єднали зусилля в створенні стандарту Plug and Play (PnP – вмикай і грай). Цей стандарт – обширний комплекс програмних і апаратних засобів із повністю автоматичним налаштуванням конфігурації комп'ютера відповідно до використовуваного ним устаткування.

Технологія PnP припускає, що досить включити комп'ютер, як всі апаратні і програмні засоби автоматично оптимально налаштуються і стануть працювати без збоїв і конфліктів.

6.8. ПЕРИФЕРІЙНІ ПРИСТРОЇ ДЛЯ СИСТЕМ СПОСТЕРЕЖЕННЯ

Підвищений інтерес кримінальних угруповань до об'єктів, що характеризуються наявністю значних грошових коштів, матеріальних або культурних цінностей (особливо це стосується банків, офісів, музеїв, складів, баз, великих універмагів і т. п.) викликав необхідність комплексного підходу до вирішення проблеми забезпечення їх безпеки, який, зокрема, має на увазі застосування інтегрованих засобів охорони, які включають засоби відеоконтролю, охоронно-пожежної сигналізації, контролю доступу та інші інженерні засоби захисту, об'єднані загальною системою управління і призначені для спільної роботи.

Функціональні можливості і ефективність комп'ютерних технічних засобів відеоспостереження (ТЗВ) якнайкраще проявляються при організації з їх допомогою інтегрованих систем охорони.

Відеоквадратори

Відеоквадратори – це цифрові пристрої, що забезпечують розміщення зображень від 4 відеоджерел на одному екрані, який у цьому випадку ділиться на 4 частини (квадранти). Відеоквадратори дозволяють зменшити кількість моніторів у системі. Існують також квадратори з більш високим розподілом, які дозволяють працювати на одному моніторі з 8 камерами: вони формують дві групи по 4 камери і дають можливість по черзі виводити їх на екран. Розрізняють відеоквадратори «реального часу», що забезпечують одночасну зміну зображень в усіх 4 квадрантах, і відеоквадратори послідовного типу зміни зображень. Більшість квадраторів можуть працювати як комутатор послідовної дії, тобто підключати будь-яку з працюючих камер до монітора. Квадратори для систем спостереження мають додаткові (за кількістю камер) тривожні входи для підключення засобів сигналізації і забезпечують виведення зображення з камери на повний екран при спрацьовуванні в її зоні засобів сигналізації. Також вони мають режим «заморожування»

кадру, тобто можливість зафіксувати зображення в одному з сегментів, передання сигналу тривоги іншим споживачам і у разі необхідності запис на відеомагнітофон. Відеоквадратори, як і відеокомутатори послідовної дії, є порівняно простими пристроями і застосовуються, як правило, у невеликих, недорогих системах.

Відеодетектори руху

Відеодетектор руху є електронним блоком, який зберігає в пам'яті поточне зображення з телекамери і подає сигнал тривоги при виникненні змін у зоні, що охороняється. Відеодетектори руху застосовуються головним чином у системах охорони великих об'єктів, де операторові доводиться контролювати велику кількість камер. Розрізняють аналогові і цифрові детектори руху. Найбільш простими і дешевими є аналогові детектори, дію яких можна при деяких припущеннях порівняти з дією охоронних сповіщувачів, що підключаються до тривожних входів комутаторів, квадраторів і тому подібне. Цифрові відеодетектори руху – це багатоканальні пристрої, які дозволяють розбивати кожну зону, що охороняється, на окремі блоки, для кожного з яких встановлюється свій поріг спрацьовування – чим вище цей поріг, тим більші зміни повинні статися на «картинці». Окрім цього, характеристики руху (початок руху, напрям, швидкість та ін.) можна визначати програмним шляхом. Це дозволяє, наприклад, не сприймати людину, що рухається в напрямі від об'єкта, що охороняється, або паралельно йому на деякій безпечній відстані, як порушника. Налаштування системи з цифровими детекторами на оптимальний режим проводиться з урахуванням особливостей місця, де встановлені телекамери, та характеристик об'єкта, що охороняється, (вірогідні шляхи переміщення порушника, наявність вразливих місць та ін.) інакше важко уникнути великої кількості помилкових спрацьовувань або, навпаки, пропуску порушника. Цифрові відеодетектори руху застосовуються в складних системах спостереження високого класу.

Відеомультіплексори

Відеомультіплексори є високотехнологічними системами відеозапису і управліннями, що мають широкі функціональні можливості і призначені для запису відеосигналів від декількох (до 16) камер на одну відеокасету (за допомогою кодування), відтворення кодованих касет і оброблення сигналів тривоги. Мультіплексори дозволяють здійснювати перемикання між різними методами запису, що дає можливість або записувати те, що з'являється на екрані, або переглядати на екрані зображення від одних камер, записуючи в цей же час зображення від інших камер. Завдяки наявності декількох режимів виведення

зображень на екран записані зображення можуть бути відтворені на одному моніторі в повноекранному режимі, режимах квадрованого екрана і «картинка в картинці» або в мультиекранному режимі. Для детальнішого аналізу повноекранних зображень багато мультиплексорів мають функцію двократного цифрового збільшення зображення. Деякі мультиплексори мають вбудовані відеодетектори руху, генератори титрів, дати і часу, а також можуть працювати в дуплексному режимі, тобто дозволяють переглядати раніше зроблені записи одночасно з поточним записом зображень із працюючих телекамер. Широкий набір вбудованих функцій, розвинена логіка оброблення сигналів тривоги, а також можливість програмування відеомультимплексорів за допомогою функціональних клавіш або з персонального комп'ютера дозволяють створювати на їх базі середні і великі (з обслуговуванням до 128 або 256 камер) телевізійні системи відеоконтролю, для чого провідними фірмами розроблений цілий спектр додаткової апаратури: адаптери видаленої клавіатури, багатопортові контролери, системи телеметричного управління камерами та ін.

Матричні відеокомутатори

Матричні комутатори мають вбудований процесор і забезпечують незалежну комутацію відеосигналів із великою кількістю входів на будь-який із моніторів. За наявності детектора руху комутатор самостійно відстежує ситуацію і в разі тривоги виводить зображення саме того приміщення, де спрацювала сигналізація, а також видає звуковий сигнал для привертання уваги оператора. Матричні комутатори дозволяють формувати декілька послідовностей зображень від камер у будь-якому порядку з управлінням їх поворотними пристроями і варіооб'єктивами, а також виводити номери камер і назви приміщень, у яких вони встановлені, повідомляти про сигнали тривоги, поточний час, дату, інструкції операторові та ін. Матричні комутатори є основними елементами багатьох систем спостереження, оскільки дозволяють створювати гнучкі системи безпеки.

Пристрої реєстрації інформації

Спеціалізовані відеомагнітофони

Спеціалізовані відеомагнітофони призначені для реєстрації і документування впродовж тривалого часу подій, що відбуваються в зонах, що охороняються. Відеомагнітофони можуть працювати у двох режимах: безперервному (час запису на стандартну відеокасету 180 хв) і переривчастому (час запису 24, 480 або 960 год). У переривчастому режимі записуються не всі кадри, а тільки визначені (див. табл. 6.5).

Режими запису на одну касету

Тривалість запису, в год	Кадри, які записуються	Кількість кадрів за N секунд
3	усі кадри	25/1
24	кожен 8-й кадр	3/1
480	кожен 160-й кадр	1/7
960	кожен 320-й кадр	1/14

При документуванні відеозапису повинен використовуватися генератор дати-часу, за допомогою якого визначається поточний час доби і дата. Важливими характеристиками відеомагнітофона є його роздільна здатність і надійність. Висока роздільна здатність дозволяє зафіксувати навіть дрібні деталі, а надійність важлива тому, що такі відеомагнітофони призначені для безперервної роботи впродовж декількох років.

Відеопринтери

Відеопринтери призначені для оперативного роздруку вибраного кадру від джерела відеосигналу. Основними характеристиками відеопринтерів є роздільна здатність, розмір знімка і можливість багатокадрового друку.

Пристрої передавання телевізійного сигналу*Канали передавання телевізійного сигналу*

Для передавання телевізійного сигналу в системах спостереження можуть використовуватися як дротові канали зв'язку (коаксіальні кабелі, телефонні лінії, волоконно-оптичні лінії), так і бездротові канали – радіоканал або інфрачервоний канал. Найбільш стабільна і якісна робота системи можлива тільки при використанні коаксіальних кабелів. Основними характеристиками кабелю є його хвильовий опір, діаметр і погонне загасання. Як правило, входні і вихідні опори основних компонентів мають значення 75 Ом, тобто розраховані на застосування кабелів із хвильовим опором 75 Ом, тому застосовувати для передавання відеосигналу кабелі з хвильовим опором 50 Ом не треба. Особливу увагу слід приділяти вибору коаксіального кабелю для зовнішнього використання (на вулиці, у неопалювальних приміщеннях, у приміщеннях з агресивним середовищем і т. п.) Ці кабелі повинні працювати в широкому діапазоні температур (± 50 °C), бути стійкими до дій сонячного світла, радіації, агресивних середовищ (у тому числі землі), мати броньований захисний екран від механічних ушкоджень. Окрім цього, необхідно врахувати, що розводка таких кабелів повинна проводитися в спеціальних клемних або розподільних коробках. Зручним є застосування спеціальних кабелів, у яких коаксіальний кабель

поєднаний із дротами живлення, або в яких по одному коаксіальному кабелю передаються живлення, відеосигнал і синхроімпульси. При необхідності передання сигналу на великі відстані застосовують відеопідсилювачі і модеми (передавачі-модулятори і приймачі-демодулятори). При цьому відеосигнал за допомогою спеціальної апаратури перетворюється, запам'ятовується і передається з використанням модему. Час передання може складати від долей секунди до хвилини (залежно від вимог до якості «картинки»). Нині найширше використовуються три системи передання зображень по цифрових і звичайних телефонних лініях:

1) системи з компресією зображень за принципом «умовного оновлення» (CR), призначені для передання тільки інформації про зміну зображення від кадру до кадру;

2) системи з MPEG-компресією, у яких використовують спеціальні алгоритми компресії зображень об'єктів, що рухаються;

3) системи з JPEG-компресією, які забезпечують незалежне стискування кадру зображення.

У спеціальних системах спостереження, коли необхідна підвищена завадозахищеність, конфіденційність інформації і висока роздільна здатність, застосовуються волоконно-оптичні лінії зв'язку. Дальність дії таких систем (як і при переданні по телефонних лініях) практично не обмежена. Відносна дорожнеча таких систем обумовлена тим, що відеокамери не мають виходу для підключення оптоволоконного кабелю, тому в систему додаються перетворювачі електричного сигналу в оптичний і назад. Окрім цього, прокладка, з'єднання і підключення досить складні.

Проте розвитку волоконно-оптичних систем останнім часом приділяється підвищена увага, особливо з боку банківських структур. При створенні мобільних і переносних систем, а також якщо прокладення кабельних ліній неможливе або недоцільне, використовується радіо- або інфрачервоний канали зв'язку. Дальність передання при цьому складає від кількох сотень метрів до кількох кілометрів. У простому випадку камера підключається до радіопередавача дециметрового діапазону, а сигнал приймається на звичайний телевізор. Проте такі системи мають істотні недоліки: можуть створювати перешкоди побутовому телемовленню, а сигнал в зоні дії передавача може приймати злочинець. Цих недоліків позбавлені радіосистеми, що працюють у сантиметровому діапазоні, а також інфрачервоні системи. Останні не вимагають дозволу на застосування від Державного комітету з радіочастот, проте вони працюють тільки в зоні прямої видимості, а їх дальність дії значною мірою

залежить від оптичної прозорості середовища (сніг, дощ, туман, пил і т. п.).

Питання для самоперевірки

1. Назвіть основні складові апаратури персонального комп'ютера.
2. Назвіть додаткові пристрої, які можна підключити до системного блока.
3. Назвіть внутрішні пристрої системного блока.
4. У чому полягає різниця між системною й об'єднувальною платами?
5. Назвіть стандартні розміри системної плати.
6. Які існують класи НВІС?
7. Назвіть види пам'яті.
8. Охарактеризуйте основні види мікросхем оперативної пам'яті.
9. Назвіть основні параметри візуальних засобів виведення даних.
10. Які існують типи засобів візуального виведення даних?
11. Назвіть стандарти безпеки, яким повинні відповідати засоби візуального виведення даних.
12. Чому розрізняють моделі кольороутворення для засобів візуального виведення даних і принтерів?
13. Охарактеризуйте типи технологій для кольорового друку.
14. Назвіть основні категорії клавіатури залежно від механізму, який використовується при роботі клавіш.
15. Дайте характеристику пристроїв зберігання даних.
16. Охарактеризуйте базове мультимедійне апаратне забезпечення.
17. Охарактеризуйте мультимедійне програмне забезпечення.
18. Наведіть приклади схем стиснення даних і поясніть їх сутність.
19. Які функції виконує аудіоадаптер?
20. Назвіть способи синтезування музичних звуків.
21. У чому полягає сутність технології WT-синтезу?
22. Для чого використовується цифровий сигнальний процесор DSP?
23. Наведіть приклади основних способів запису інформації на лазерні диски.
24. У чому полягає сутність способу запису інформації з постійною кутовою швидкістю.
25. Назвіть й охарактеризуйте типи цифрових дисків.

РОЗДІЛ 7

АРХІТЕКТУРА КОМП'ЮТЕРНИХ МЕРЕЖ

Мережа – це сукупність об'єктів, що утворюються пристроями передавання і оброблення даних. Міжнародна організація з стандартизації визначила обчислювальну мережу як послідовне біт-орієнтоване передавання інформації між пов'язаними один з одним незалежними пристроями.

Мережі звичайно знаходяться в приватному веденні користувача і займають деяку територію. За територіальною ознакою мережі розділяються на:

1. Локальні обчислювальні мережі (ЛОМ), або *Local Area Network* (LAN), розташовані в одному або декількох близько розташованих будівлях. ЛОМ звичайно розміщуються в рамках якої-небудь організації (корпорації, установи), тому їх називають корпоративними.

2. Розподілені комп'ютерні мережі, глобальні, або *Wide Area Network* (WAN), розташовані в різних будівлях, містах і країнах, які бувають територіальними, змішаними і глобальними. Залежно від цього глобальні мережі бувають чотирьох основних видів: міські, регіональні, національні і транснаціональні. Як приклади розподілених мереж дуже великого масштабу можна назвати: Internet, EUNET, Relcom, FIDO.

До складу мережі в загальному випадку включаються наступні елементи:

- мережеві комп'ютери (оснащені мережевим адаптером);
- канали зв'язку (кабельні, супутникові, телефонні, цифрові, волоконно-оптичні, радіоканали та ін.);
- різного роду перетворювачі сигналів;
- мережеве устаткування.

Розрізняють два поняття мережі: комунікаційна мережа і інформаційна мережа (див. рис. 7.1).

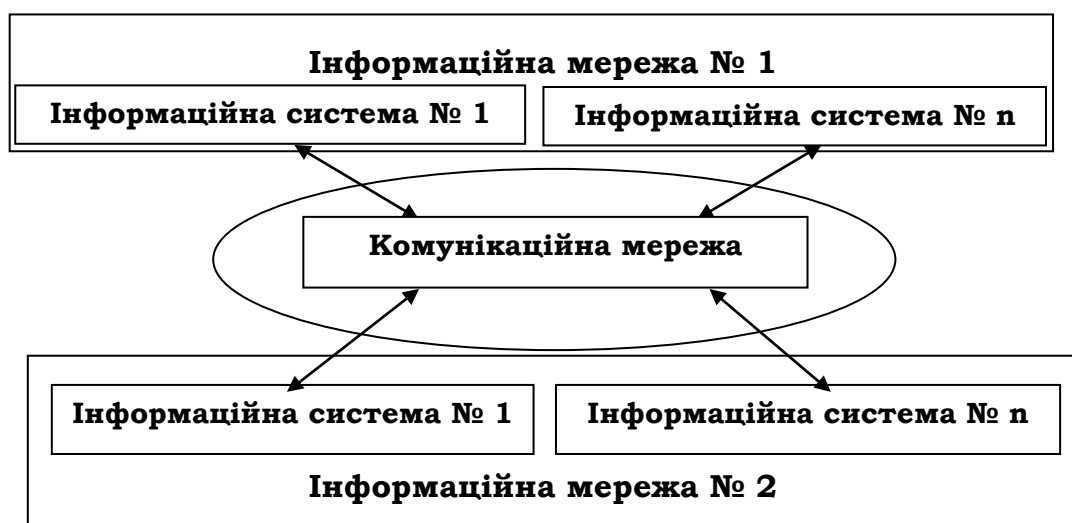


Рис. 7.1. Інформаційні і комунікаційні мережі

Комунікаційна мережа призначена для передання даних, також вона виконує завдання, пов'язані з перетворенням даних. Комунікаційні мережі розрізняються за типом використовуваних фізичних засобів з'єднання.

Інформаційна мережа призначена для зберігання інформації і складається з інформаційних систем. На базі комунікаційної мережі може бути побудована група інформаційних мереж.

Під інформаційною системою слід розуміти систему, яка є постачальником або споживачем інформації.

Комп'ютерна мережа складається з інформаційних систем і каналів зв'язку.

Під інформаційною системою слід розуміти об'єкт, здатний здійснювати зберігання, оброблення та передання інформації. До складу інформаційної системи входять: комп'ютери, програми, користувачі та інші складові, призначені для процесу оброблення і передання даних. Надалі інформаційна система, призначена для вирішення завдань користувача, називатиметься робоча станція (*client*). Робоча станція в мережі відрізняється від звичайного персонального комп'ютера (ПК) наявністю мережевої карти (мережевого адаптера), каналу для передання даних і мережевого програмного забезпечення.

Під каналом зв'язку слід розуміти шлях або засіб, по якому передаються сигнали. Засіб передавання сигналів називають абонентським, або фізичним, каналом.

Канали зв'язку (*data link*) створюються по лініях зв'язку за допомогою мережевого устаткування і фізичних засобів зв'язку. Фізичні засоби зв'язку побудовані на основі витих пар, коаксіальних кабелів, оптичних каналів або ефіру. Між взаємодіючими інформаційними системами через фізичні канали комунікаційної мережі і вузли комутації встановлюються логічні канали.

Логічний канал – це шлях для передання даних від однієї системи до іншої. Логічний канал прокладається по маршруту в одному або декількох фізичних каналах. Логічний канал можна охарактеризувати як маршрут, прокладений через фізичні канали і вузли комутації.

Інформація в мережі передається блоками даних за процедурами обміну між об'єктами. Ці процедури називають протоколами передавання даних.

Протокол – це сукупність правил, що встановлюють формат і процедури обміну інформацією між двома або декількома пристроями.

Завантаження мережі характеризується параметром, що називається трафіком. Трафік (*traffic*) – це потік повідомлень у

мережі передавання даних. Під ним розуміють кількісне вимірювання у вибраних точках мережі числа блоків даних і їх довжини, що проходять, виражене в бітах за секунду.

Істотний вплив на характеристики мережі надає метод доступу. Метод доступу – це спосіб визначення того, яка з робочих станцій зможе використовувати канал зв'язку і як управляти доступом до каналу зв'язку (кабелю).

У мережі всі робочі станції фізично сполучені між собою каналами зв'язку за певною структурою, яка називається топологією. Топологія – це опис фізичних з'єднань у мережі, що вказує, які робочі станції можуть зв'язуватися між собою. Тип топології визначає продуктивність, працездатність і надійність експлуатації робочих станцій, а також час звернення до файлового сервера. Залежно від топології мережі використовується той або інший метод доступу.

Склад основних елементів у мережі залежить від її архітектури. Архітектура – це концепція, що визначає взаємозв'язок, структуру і функції взаємодії робочих станцій у мережі. Вона передбачає логічну, функціональну і фізичну організацію технічних і програмних засобів мережі. Архітектура визначає принципи побудови і функціонування апаратного і програмного забезпечення елементів мережі.

Сучасні мережі можна класифікувати за різними ознаками: за віддаленістю комп'ютерів, топологією, призначенням, переліком послуг, що надаються, принципами управління (централізовані і децентралізовані), методами комутації, методами доступу, видами середовища передавання, швидкостями передавання даних та ін.

Переваги використання мереж

Комп'ютерні мережі є варіантом співпраці людей і комп'ютерів, що забезпечує прискорення доставки і оброблення інформації. Коли можливості комп'ютерів вирости і ПК стали доступні кожному, розвиток мереж значно прискорився.

Сполучені в мережу комп'ютери обмінюються інформацією і спільно використовують периферійне устаткування і пристрої зберігання інформації (див. рис. 7.2).

За допомогою мереж можна розділяти ресурси і інформацію. Нижче перераховані основні завдання, які розв'язуються за допомогою робочої станції в мережі і які важко вирішити за допомогою окремого комп'ютера:

1. Комп'ютерна мережа дозволить спільно використовувати периферійні пристрої, включаючи:

- принтери;
- плотери;

- дискові накопичувачі;
- приводи CD-ROM;
- дисководи;
- стримери;
- сканери;
- факс-модеми.

2. Комп'ютерна мережа дозволяє спільно використовувати інформаційні ресурси:

- каталоги;
- файли;
- прикладні програми;
- ігри;
- бази даних;
- текстові процесори.

3. Комп'ютерна мережа дозволяє працювати з розрахованими на багато користувачів програмами, що забезпечують одночасний доступ всіх користувачів до загальних баз даних з блокуванням файлів і записів, що забезпечує цілісність даних. Будь-які програми, розроблені для стандартних ЛОМ, можна використовувати в інших мережах.

Спільне використання ресурсів забезпечить істотну економію засобів і часу. Наприклад, можна колективно використовувати один лазерний принтер замість покупки принтера кожному співробітнику або метушні з дискетами до єдиного принтера за відсутності мережі.

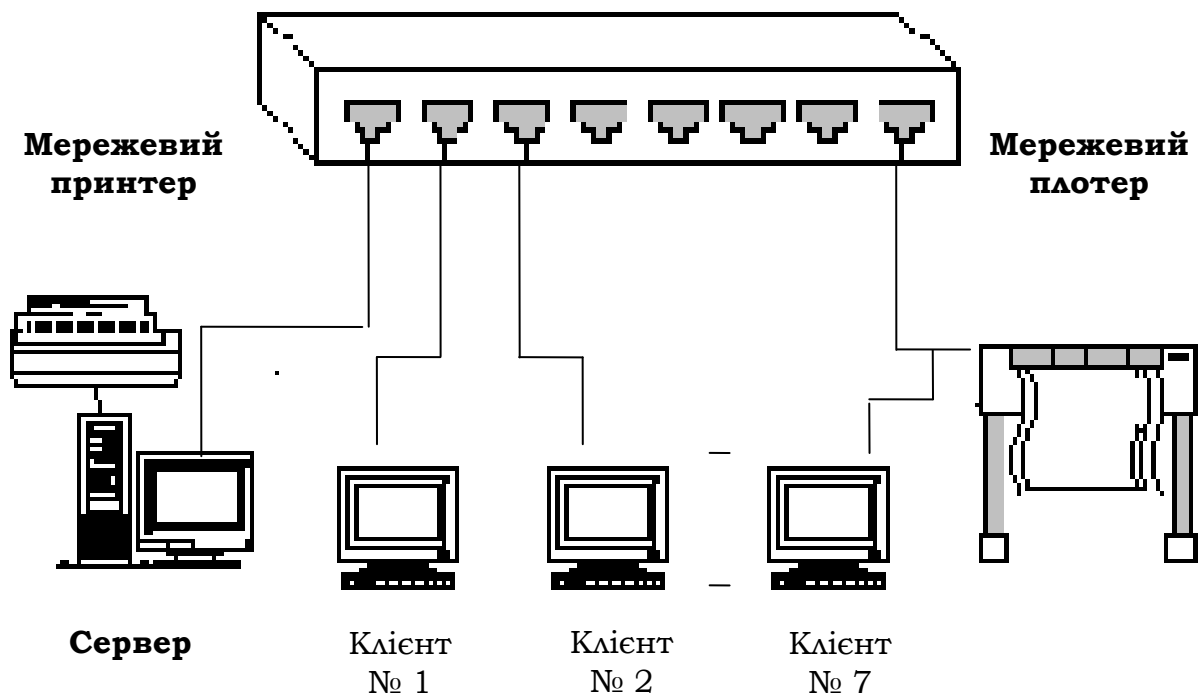


Рис. 7.2. Використання периферійного устаткування

Організація електронної пошти. Можна використовувати ЛОМ як поштову службу і розсилати службові записки, доповіді і повідомлення іншим користувачам.

Архітектура мережі визначає основні елементи мережі, характеризує її загальну логічну організацію, технічне забезпечення, програмне забезпечення, описує методи кодування. Архітектура також визначає принципи функціонування і інтерфейс користувача.

Розглянемо три види архітектури:

- 1) архітектура «термінал – головний комп'ютер»;
- 2) однорангова архітектура;
- 3) архітектура «клієнт – сервер».

7.1. АРХІТЕКТУРА «ТЕРМІНАЛ – ГОЛОВНИЙ КОМП'ЮТЕР»

Архітектура «термінал – головний комп'ютер» (*terminal – host computer architecture*) – це концепція інформаційної мережі, у якій все оброблення даних здійснюється одним або групою головних комп'ютерів.

Головний комп'ютер

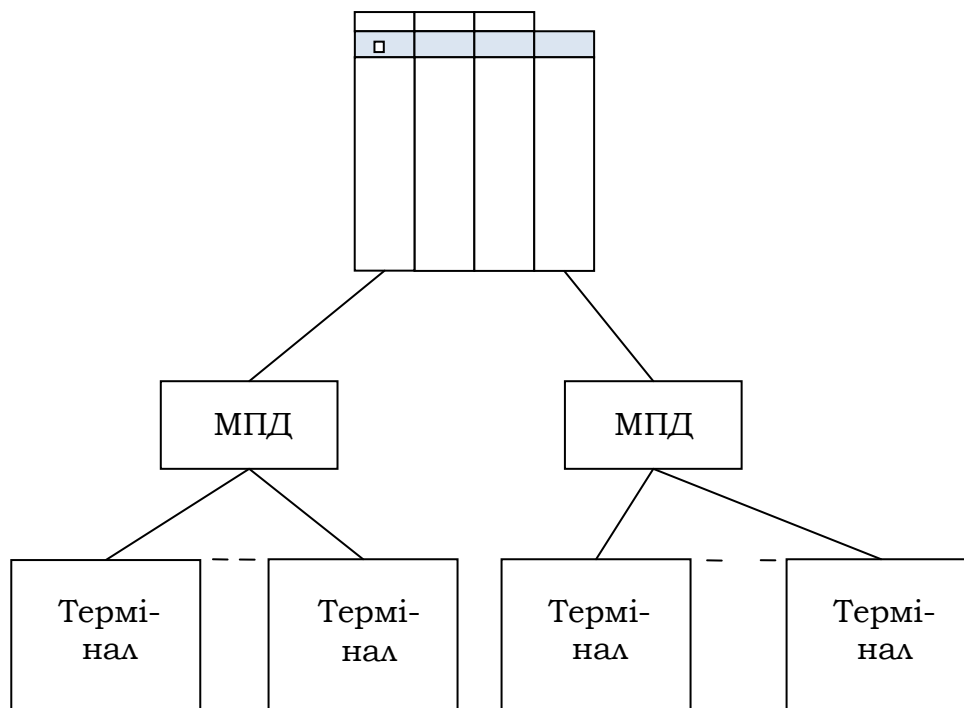


Рис. 7.3. Архітектура «термінал – головний комп'ютер»

Дана архітектура припускає два типи устаткування:

- головний комп'ютер, де здійснюється управління мережею, зберігання і оброблення даних;
- термінали, призначені для передання головному комп'ютеру команд на організацію сеансів і виконання завдань, введення даних для виконання завдань і отримання результатів.

Головний комп'ютер через мультиплексори передавання даних (МПД) взаємодіють із терміналами, як представлено на рис. 7.3.

Класичний приклад архітектури мережі з головними комп'ютерами – системна мережева архітектура (*System Network Architecture* – SNA).

7.2. ОДНОРАНГОВА АРХІТЕКТУРА

Однорангова архітектура (*peer-to-peer architecture*) – це концепція інформаційної мережі, у якій її ресурси розосереджені по всіх системах (див. рис. 7.4). Дана архітектура характеризується тим, що в ній всі системи рівноправні.

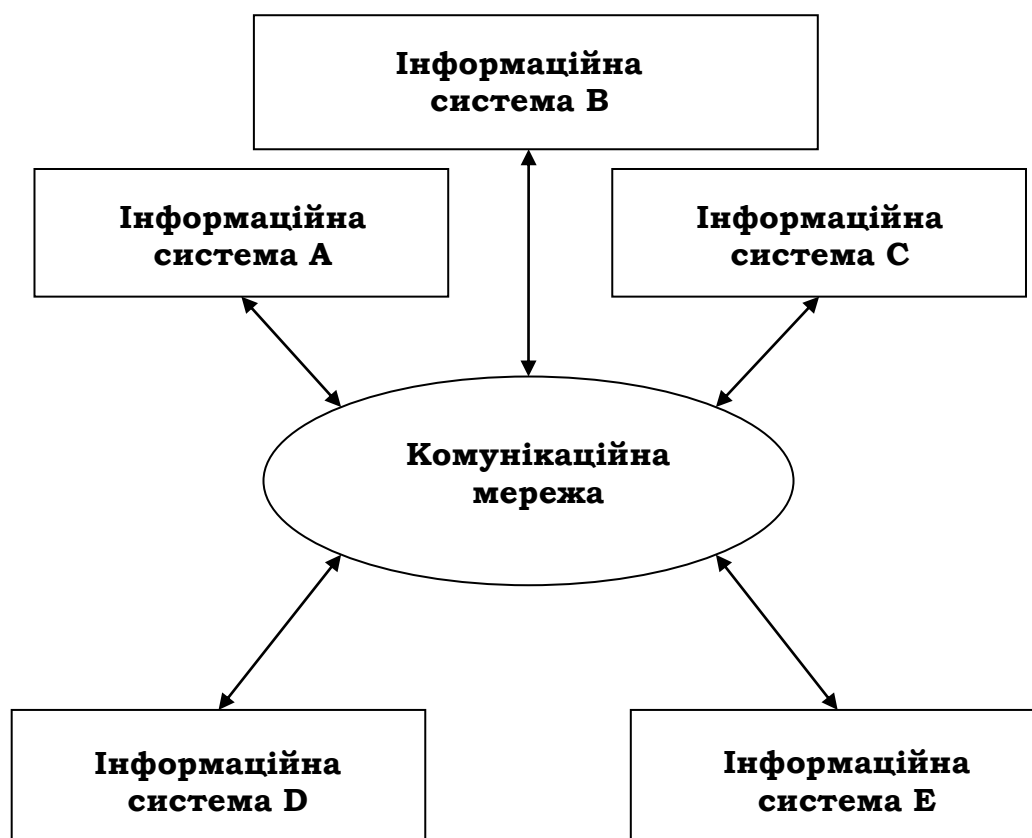


Рис. 7.4. Однорангова архітектура

До однорангових мереж відносяться малі мережі, де будь-яка робоча станція може виконувати одночасно функції файлового сервера і робочої станції. У однорангових ЛОМ дисковий простір і файли на будь-якому комп'ютері можуть бути спільними. Щоб ресурс став спільним, його необхідно віддати в спільне користування, використовуючи служби видаленого доступу мережеских однорангових операційних систем. Залежно від того, як буде встановлений захист даних, інші користувачі зможуть користуватися файлами відразу ж після їх створення. Однорангові ЛОМ достатньо хороші тільки для невеликих робочих груп.

Однорангові ЛОМ є найбільш легким і дешевим типом мереж для установки. Вони на комп'ютері вимагають, окрім мережевої карти і мережевого носія, тільки операційної системи Windows 95 або Windows for Workgroups. При з'єднанні комп'ютерів користувачі можуть надавати ресурси і інформацію в спільне користування.

Однорангові мережі мають наступні переваги:

- легкі в установці і налаштуванні;
- окремі ПК не залежать від виділеного сервера;
- користувачі в змозі контролювати свої ресурси;
- мала вартість і легка експлуатація;
- мінімум устаткування і програмного забезпечення;
- немає необхідності в адміністраторі;
- добре підходять для мереж із кількістю користувачів, що не перевищує десяти.

Проблемою однорангової архітектури є ситуація, коли комп'ютери відключаються від мережі. У цих випадках з мережі зникають види сервісу, які вони надавали. Мережеву безпеку одночасно можна застосувати тільки до одного ресурсу, і користувач повинен пам'ятати стільки паролів, скільки мережевих ресурсів. При отриманні доступу до ресурсу, що розділяється, відчувається падіння продуктивності комп'ютера. Істотним недоліком однорангових мереж є відсутність централізованого адміністрування.

Використання однорангової архітектури не виключає застосування в тій же мережі також архітектури «термінал – головний комп'ютер» або архітектури «клієнт – сервер».

7.3. АРХІТЕКТУРА «КЛІЄНТ – СЕРВЕР»

Архітектура «клієнт – сервер» (*client-server architecture*) – це концепція інформаційної мережі, у якій основна частина її ресурсів зосереджена в серверах, що обслуговують своїх клієнтів (див. рис. 7.5).

Дана архітектура визначає два типи компонентів: сервери і клієнти.

Сервер – це об'єкт, що надає сервіс іншим об'єктам мережі на їх запити. Сервіс – це процес обслуговування клієнтів.

Сервер працює за завданнями клієнтів і управляє виконанням їх завдань. Після виконання кожного завдання сервер надсилає отримані результати клієнту, що поставив це завдання.

Сервісна функція в архітектурі «клієнт – сервер» описується комплексом прикладних програм, відповідно до якого виконуються різноманітні прикладні процеси.

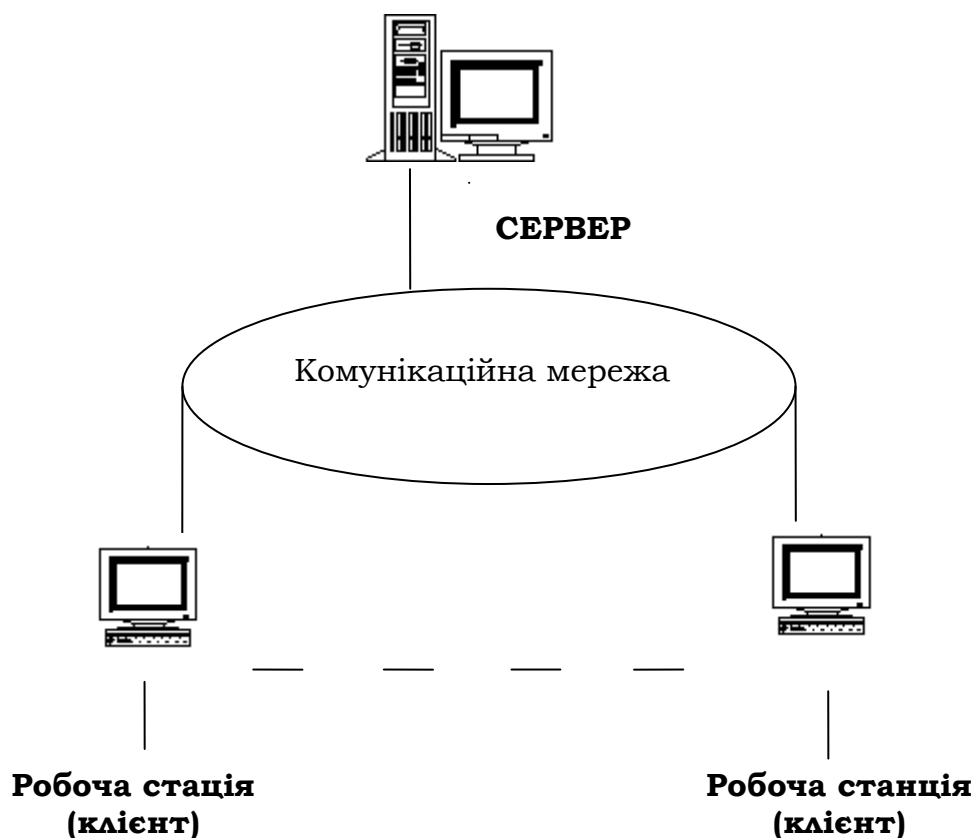


Рис. 7.5. Архітектура «клієнт – сервер»

Процес, який викликає сервісну функцію за допомогою певних операцій, називається клієнтом. Ним може бути програма або користувач. На рис. 7.6 наведений перелік сервісів в архітектурі «клієнт – сервер».

Клієнти – це робочі станції, які використовують ресурси сервера і надають зручні інтерфейси користувача. Інтерфейси користувача – це процедури взаємодії користувача з системою або мережею.

Клієнт є ініціатором і використовує електронну пошту або інші сервіси сервера. У цьому процесі клієнт запрошує вид обслуговування, встановлює сеанс, одержує потрібні йому результати і повідомляє про закінчення роботи.

У мережах з виділеним файловим сервером на виділеному автономному ПК встановлюється серверна мережева операційна система. Цей ПК стає сервером. Програмне забезпечення (ПЗ), встановлене на робочій станції, дозволяє їй обмінюватися даними з сервером. Найбільш поширені мережеві операційні системи:

- NetWare фірми Novel;
- Windows NT фірми Microsoft;
- UNIX фірми AT&T;
- Linux.

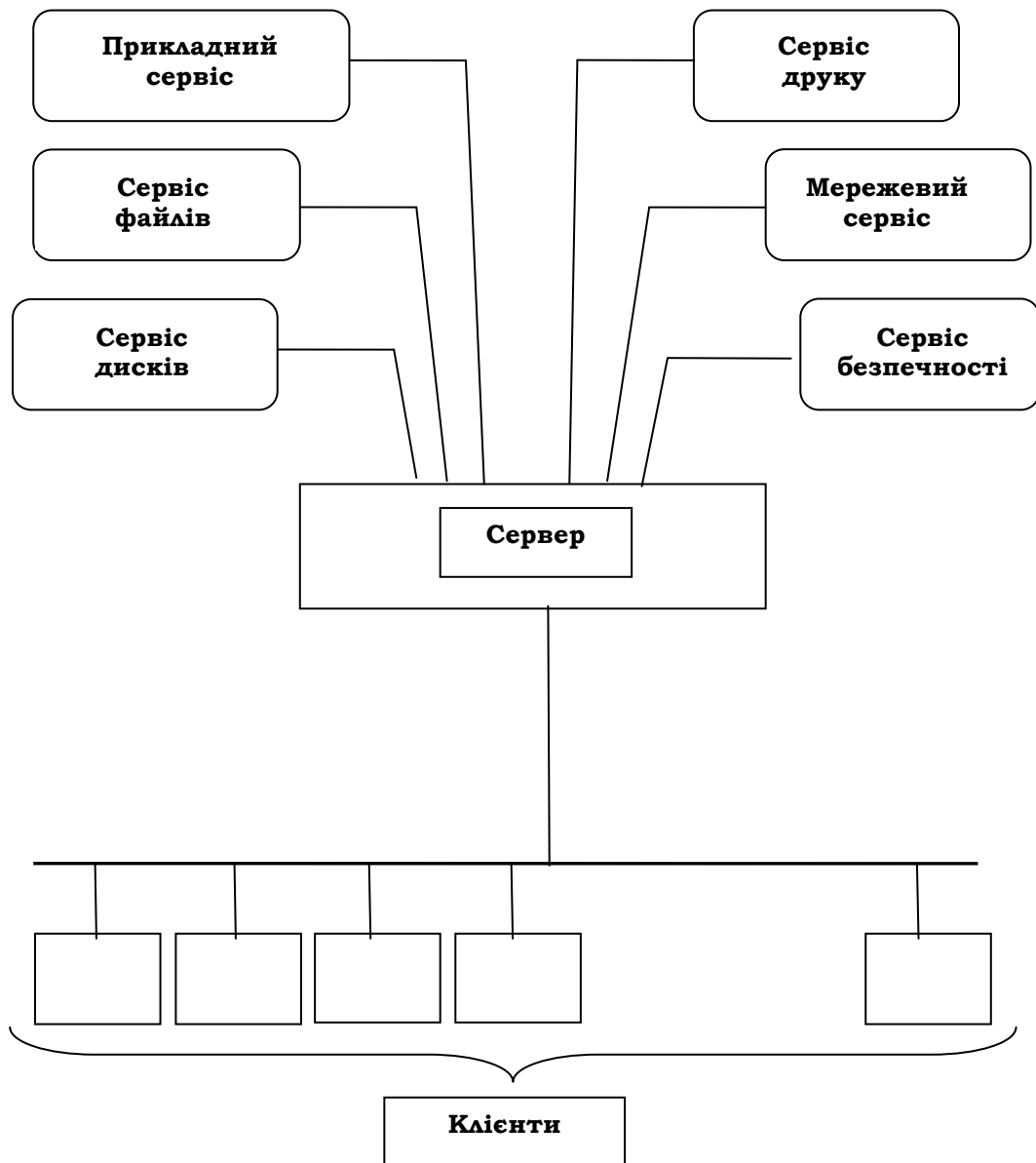


Рис. 7.6. Модель сервер-клієнта

Крім мережевої операційної системи необхідні мережеві прикладні програми, що реалізують переваги, які надаються мережею.

Мережі на базі серверів мають кращі характеристики і підвищену надійність. Сервер володіє головними ресурсами мережі, до яких звертається решта робочих станцій.

У сучасній клієнт-серверній архітектурі виділяється чотири групи об'єктів: клієнти, сервери, дані і мережеві служби. Клієнти розташовуються в системах на робочих місцях користувачів. Дані в основному зберігаються в серверах. Мережеві служби спільно використовуються серверами і даними. Крім того, служби управляють процедурами оброблення даних.

Мережі клієнт-серверної архітектури мають наступні переваги:

- дозволяють організовувати мережі з великою кількістю робочих станцій;

- забезпечують централізоване управління обліковими записами користувачів, безпекою і доступом, що спрощує мережеве адміністрування;

- ефективний доступ до мережевих ресурсів;
- користувачу потрібен один пароль для входу в мережу і для отримання доступу до всіх ресурсів, на які поширюються права користувача.

Разом із перевагами мережі клієнт-серверної архітектури мають і низку недоліків:

- несправність сервера може зробити мережу непридатною, наприклад через втрату мережевих ресурсів;
- вимагає кваліфікованого персоналу для адміністрування;
- має вищу вартість мереж і мережевого устаткування.

Вибір архітектури мережі залежить від призначення мережі, кількості робочих станцій і від виконуваних на ній дій.

Слід вибрати однорангову мережу, якщо:

- кількість користувачів не перевищує десяти;
- усі машини знаходяться близько одна від одної;
- фінансові можливості невеликі;
- немає необхідності в спеціалізованому сервері, такому як сервер БД, факс-сервер або який-небудь інший;
- немає можливості або необхідності в централізованому адмініструванні.

7.4. БАЗОВІ ТОПОЛОГІЇ МЕРЕЖ

Термін «топология», чи «топология мережі», характеризує фізичне розташування комп'ютерів, кабелів і інших компонентів мережі. Топология – це стандартний термін, що використовується професіоналами для опису формування мережі. Крім терміна «топология», для опису фізичного компонування вживають також наступні:

- фізичне розташування;
- компонування;
- діаграма;
- карта.

Топология мережі обумовлює її характеристики. Зокрема, вибір тієї чи іншої топологии впливає на:

- склад необхідного мережевого устаткування;
- характеристики мережевого устаткування;
- можливості розширення мережі;
- засіб керування мережею.

Щоб спільно використовувати ресурси або виконувати інші мережеві задачі, комп'ютери повинні бути підключені один до одного. Для цієї мети в більшості мереж застосовується кабель чи використовується один з безпроводних засобів зв'язку.

Однак просто підключити комп'ютер до кабелю, що з'єднує інші комп'ютери, не досить. Різні типи кабелів у сполученні з різними мережевими платами, мережевими операційними системами та іншими компонентами вимагають і різного взаємного розташування комп'ютерів.

Кожна топологія мережі накладає ряд умов. Наприклад, вона може диктувати не тільки тип кабелю, але й засоби його прокладки. Топологія може також визначати спосіб взаємодії комп'ютерів у мережі. Різним видам топологій відповідають різні методи взаємодії, і ці методи дуже впливають на мережу.

Усі мережі будуються на основі трьох базових топологій:

- шина (*bus*);
- зірка (*star*);
- кільце (*ring*).

Якщо комп'ютери підключені уздовж одного кабелю (сегмента (*segment*)), то топологія називається **шиною**. У тому випадку, коли комп'ютери підключені до сегментів кабелю, що виходить з однієї точки, чи концентратора, топологія називається **зіркою**. Якщо кабель, до якого підключені комп'ютери, замкнений у кільце, така топологія зветься **кільце**.

Хоча самі по собі базові топології нескладні, у реальності часто зустрічаються досить складні комбінації, що поєднують властивості кількох топологій.

7.5. ПРИНЦИП ФУНКЦІОНУВАННЯ ШИННОЇ ТОПОЛОГІЇ

Шинна топологія часто застосовується в невеликих, простих чи тимчасових мережових інсталяціях. Організація мережі із шинною топологією представлена на рисунку 7.7.

У типовій мережі із шинною топологією кабель містить одну чи більше пар провідників, а активні схеми посилення чи сигналу передавання його від одного комп'ютера до іншого відсутні. Таким чином, шинна топологія є пасивною. Коли одна машина надсилає сигнал по кабелю, усі інші вузли одержують цю інформацію, але тільки один із них (адреса, якого збігається з адресою, яка закодована у повідомленні) приймає її. Інші комп'ютери повідомлення не сприймають.

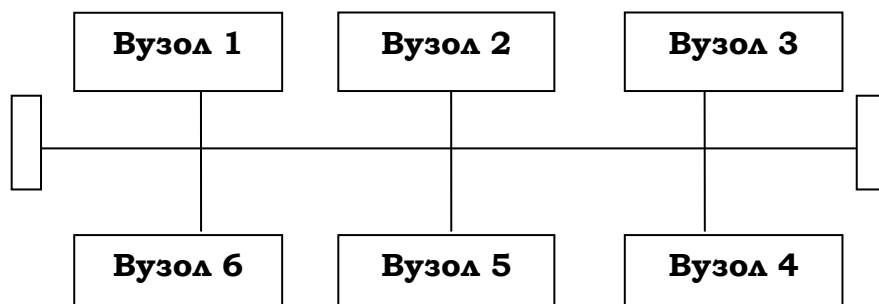


Рис. 7.7. Комп'ютерна мережа із шинною топологією

У кожен момент часу відправляти повідомлення може тільки один комп'ютер, тому число підключених до мережі машин значно впливає на її швидкість. Перед переданням даних комп'ютер повинен очікувати звільнення шини. Зазначені фактори діють також у кільцевій і зіркоподібній мережах.

Ще одним важливим фактором є кінцеве навантаження. Оскільки шинна топологія є пасивною, електричний сигнал від комп'ютера, який веде передання, вільно подорожує по всій довжині кабелю. Без кінцевого навантаження сигнал досягає кінця кабелю, відбивається і йде у зворотному напрямку. Таке відображення називається зацикленням (*ringing*). Для запобігання подібного явища до обох кінців кабельного сегмента підключається кінцеве навантаження (термінатор). Термінатори поглинають електричний сигнал і запобігають його відображенню. У мережах із шинною топологією кабелі не можна залишати без кінцевого навантаження.

Прикладом недорогої мережі із шинною топологією є Ethernet 10Base2 (таку мережу називають також «тонкою» Ethernet).

Якщо в мережі із шинною топологією виникають проблеми, переконайтеся, що до кабелю підключене кінцеве навантаження. У мережах без кінцевого навантаження з'являються часті помилки. У результаті мережі стають некерованими.

Переваги шинної топології

Шинна топологія має кілька переваг:

1. Вона надійно працює в невеликих мережах, проста у використанні і зрозуміла.
2. Шина вимагає менше кабелю для з'єднання комп'ютерів і тому дешевше ніж інші схеми кабельних з'єднань.
3. Шинну топологію легко розширити. Два кабельних сегменти можна зістикувати в один довгий кабель за допомогою циліндричного з'єднувача BNC. Це дозволяє підключити до мережі додаткові комп'ютери.
4. Для розширення мережі із шинною топологією можна використовувати повторювач. Повторювач (*repeater*) підсилює сигнал і дозволяє передавати його на великі відстані.

Недоліки шинної топології

Шинна топологія має наступні недоліки:

1. Інтенсивний мережевий трафік значно знижує продуктивність такої мережі. Оскільки будь-який комп'ютер може передавати дані в довільний момент часу і в більшості мереж вони не координують один з одним моменти передання, у мережі з шинною топологією та великим числом комп'ютерів станції часто переривають одна одну, і чимала частина смуги пропускання

губиться даремно. При додаванні комп'ютерів до мережі проблема стає більш виразною.

2. Кожен циліндричний з'єднувач послабляє електричний сигнал. Тому велика їх кількість буде перешкоджати коректному переданню інформації з шини.

3. Мережу із шинною топологією важко діагностувати. Розрив кабелю або неправильне функціонування одного з комп'ютерів може привести до того, що інші вузли не зможуть взаємодіяти один з одним. У результаті вся мережа стає непрацездатною.

7.6. ЗІРКОПОДІБНА ТОПОЛОГІЯ: ПРИНЦИП ФУНКЦІОНУВАННЯ

У топології типу «зірка» усі кабелі йдуть до комп'ютерів від центрального вузла, де вони підключаються до концентратора (*hub*). Така топологія представлена на рис. 7.8.

Зіркоподібна топологія застосовується в мережах, у яких кінцеві точки досяжні з центрального вузла. Вона добре підійде в тих випадках, коли передбачається розширення мережі і потрібна висока надійність.



Рис. 7.8. Мережа із зіркоподібною топологією

Кожен комп'ютер у мережі з топологією типу «зірка» взаємодіє з центральним концентратором, котрий передає повідомлення всім комп'ютерам (у зіркоподібній мережі із широкомовним розсиланням) або тільки комп'ютеру-адресату (у зіркоподібній комутованій мережі).

Активний концентратор регенерує електричний сигнал і надсилає його всім підключеним комп'ютерам. Такий тип концентратора часто називають багатопортовим повторювачем (*multiport repeater*). Для роботи активних концентраторів і комутаторів потрібен струм від мережі. Пасивні концентратори, наприклад комутаційна кабельна панель або комутаційний блок, діють як точка з'єднання, не підсилюючи і не регенеруючи сигнал. Струму такі пристрої не вимагають.

Для реалізації мережі з топологією типу «зірка» можна застосовувати кілька типів кабелів. Гібридний концентратор дозволяє використовувати в одній зіркоподібній мережі різні типи кабелів.

Розширювати зіркоподібну мережу можна шляхом підключення замість одного з комп'ютерів ще одного концентратора і приєднання до нього додаткових машин. Так створюється гібридна зіркоподібна мережа, аналогічна представлений на рис. 7.9.

Переваги мережі із зіркоподібною топологією

Перелічимо деякі переваги мережі з зіркоподібною топологією:

1. Така мережа допускає просту модифікацію і додавання комп'ютерів, не порушуючи іншої її частини. Досить прокласти новий кабель від комп'ютера до центрального вузла і підключити його до концентратора. Якщо можливості центрального концентратора будуть вичерпані, варто замінити його пристроєм з більшим числом портів.

2. Центральний концентратор зіркоподібної мережі зручно використовувати для діагностики. Інтелектуальні концентратори (пристрої з мікропроцесорами, доданими для повторення мережових сигналів) забезпечують також моніторинг і керування мережею.

Гібридна зіркоподібна мережа має декілька центральних точок із з'єднаннями типу «зірка» (див. рис. 7.9).

3. Відмова одного комп'ютера не обов'язково приводить до зупинення всієї мережі. Концентратор здатен виявляти відмови та ізолювати таку машину чи мережовий кабель, що дозволяє іншій мережі продовжувати роботу.

4. В одній мережі допускається застосування кількох типів кабелів (якщо їх дозволяє використовувати концентратор).

З трьох типів мереж зіркоподібна відрізняється найбільшою гнучкістю і простотою в діагностиці у випадку відмови.

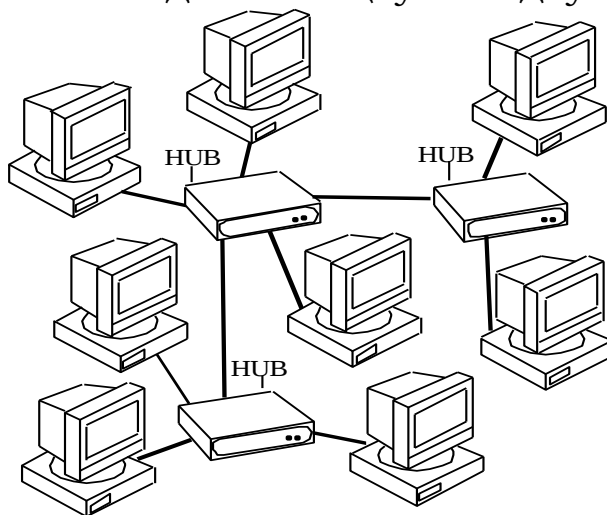


Рис. 7.9. Гібридна зіркоподібна мережа

Недоліки мережі з зіркоподібною топологією

Зіркоподібна мережа має кілька недоліків:

1. При відмові центрального концентратора стає непрацездатною вся мережа.

2. Багато мереж з топологією типу «зірка» вимагають застосування на центральному вузлі пристрою для ретрансляції широкомовних повідомлень або комутації мережевого трафіку.

3. Усі комп'ютери повинні з'єднуватися з центральною точкою, що збільшує витрати кабелю. Тому такі мережі обходяться дорожче, ніж мережі з іншою топологією.

7.7. МЕРЕЖІ З КІЛЬЦЕВОЮ ТОПОЛОГІЄЮ

У кільцевій мережі кожен комп'ютер зв'язаний із наступним, а останній – із першим (див. рис. 7.10). Кільцева топологія застосовується в мережах, що вимагають резервування визначеної частини смуги пропускання для критичних за часом засобів (наприклад, для передання відео й аудіо), у високопродуктивних мережах, а також при великій кількості клієнтів, що звертаються до мережі (що вимагає її високої пропускної здатності).

Смугою пропускання називається здатність середовища передавання даних передавати визначений обсяг інформації.

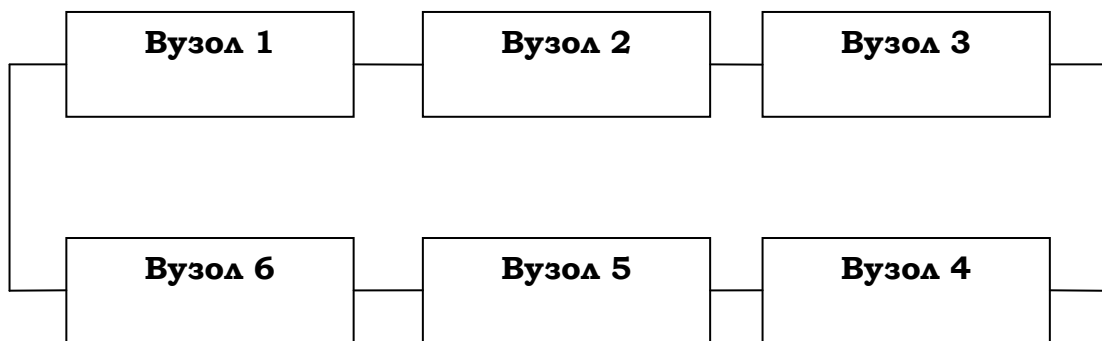


Рис. 7.10. Мережі з кільцевою топологією

Принцип роботи мереж і кільцевою топологією

У мережі з кільцевою топологією кожен комп'ютер з'єднується з наступним комп'ютером, що ретранслює ту інформацію, яку він одержує від першої машини. Завдяки такій ретрансляції мережа є активною, і в ній не виникають проблеми втрати сигналу, як у мережах із шинною топологією. Крім того, оскільки «кінця» у кільцевій мережі немає, ніяких кінцевих навантажень не потрібно.

Деякі мережі з кільцевою топологією використовують метод естафетного передавання. Спеціальне коротке повідомлення-маркер циркулює по кільцю, поки комп'ютер не побажає передати інформацію іншому вузлу. Він модифікує маркер, додає електронну адресу і дані, а потім відправляє його по кільцю. Кожний із комп'ютерів послідовно одержує даний маркер із доданою інформацією і передає його сусідній машині, поки електронна адреса не співпаде з адресою комп'ютера-одержувача чи маркер не повернеться до відправника.

Комп'ютер, що одержав повідомлення, повертає відправнику відповідь, яка підтверджує, що послання прийняте. Тоді відправник створює ще один маркер і відправляє його в мережу, що дозволяє іншій станції перехопити маркер і почати передання. Маркер циркулює по кільцю, поки яка-небудь зі станцій не буде готова до передання і не захопить його.

Усі ці події відбуваються дуже часто: маркер може пройти кільце з діаметром у 200 м приблизно 10 тис. разів за секунду. У деяких ще більш швидких мережах циркулює відразу кілька маркерів. В інших мережевих середовищах застосовуються два кільця з циркуляцією маркерів у протилежних напрямках. Така структура сприяє відновленню мережі у випадку виникнення відмовлень.

Прикладом швидкої волоконно-оптичної мережі з кільцевою топологією є FDDI (*Fiber Distributed Data Interface* – розподілений інтерфейс передання даних по волоконно-оптичним каналам: стандарт, розроблений ANSI і модифікований IEEE/ISO).

Переваги мережі з кільцевою топологією

Мережа з кільцевою топологією має наступні переваги:

1. Оскільки всім комп'ютерам дається рівний доступ до маркера, ніхто з них не зможе монополізувати мережу.
2. Справедливе спільне використання мережі забезпечує поступове зниження її продуктивності у випадку збільшення кількості користувачів і перевантаження.

Недоліки мережі з кільцевою топологією

Мережі з кільцевою топологією властиві наступні недоліки:

1. Відмова одного комп'ютера в мережі може вплинути на працездатність усієї мережі.
2. Кільцеву мережу важко діагностувати.
3. Додавання або видалення комп'ютера змушує розривати мережу.

7.8. ЗМІШАНІ ТОПОЛОГІЇ: ШИННО-ЗІРКОПОДІБНА

І ЗІРКОПОДІБНО-КІЛЬЦЕВА

Сьогодні можна зустріти мережі, у яких сполучаються шинна, зіркоподібна і кільцева топології. Дві найпоширеніші комбінації представлені на рис. 7.11.

Шинно-зіркоподібна топологія комбінує мережі типу «зірка» і «шина», зв'язуючи кілька концентраторів шинними магістралями. Якщо один з комп'ютерів відмовляється, концентратор може виявити вузол, що відмовив, і ізолювати несправну машину. При відмові концентратора з'єднані з ним комп'ютери не зможуть взаємодіяти з мережею, а шина розімкнеться на два незв'язаних один з одним сегмента (див. рис. 7.11 а).

Зіркоподібно-кільцева топологія дещо схожа на шинно-зіркоподібну. І в тій, і в іншій топології комп'ютери підключені

до концентратора, який фактично і формує кільце або шину. Відмінність в тому, що концентратори в шинно-зіркоподібній топології з'єднані магістральною лінійною шиною, а в зіркоподібно-кільцевій концентратори на основі на основі головного концентратора утворюють зірку (див. рис. 7.11 б).

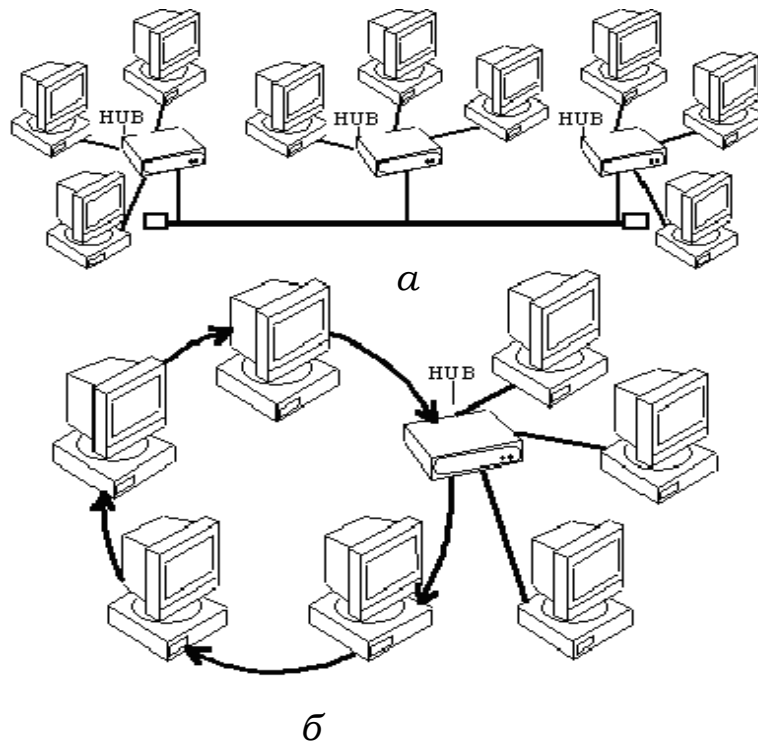


Рис. 7.11. Змішані топології:
а – шинно-зіркоподібна топологія; *б* – зіркоподібно-кільцева топологія

Питання для самоперевірки

1. Охарактеризуйте склад комп'ютерної мережі.
2. Розкрийте поняття архітектури комп'ютерної мережі.
3. Охарактеризуйте архітектуру «термінал – головний комп'ютер».
4. Охарактеризуйте однорангову архітектуру.
5. Охарактеризуйте архітектуру «клієнт – сервер».
6. Охарактеризуйте модель сервер-клієнта.
7. Визначте поняття «топологія», опишіть види базових топологій.
8. Охарактеризуйте принцип функціонування шинної топології.
9. Охарактеризуйте принцип функціонування зіркоподібної топології.
10. Охарактеризуйте принцип функціонування кільцевої топології.
11. Назвіть найпоширеніші змішані топології й охарактеризуйте їх.

РОЗДІЛ 8

ФІЗИЧНЕ СЕРЕДОВИЩЕ ПЕРЕДАННЯ

8.1. МЕРЕЖНИЙ КАБЕЛЬ

Кабель – це достатньо складний виріб, що складається з провідників, шарів екрана і ізоляції. У деяких випадках до складу кабелю входять роз'єми, за допомогою яких кабелі приєднуються до устаткування. Окрім цього, для забезпечення швидкої перекомутації кабелів і устаткування використовуються різні електромеханічні пристрої, звані кросовими секціями, кросовими коробками або шафами.

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

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

Міжнародний стандарт ISO/IEC 11801 та європейський стандарт EN50173 близькі між собою і за багатьма позиціями пред'являють до кабелів ідентичні вимоги. Проте є й відмінності між цими стандартами. Наприклад, у міжнародний стандарт ISO/IEC 11801 і європейський EN50173 увійшли деякі типи кабелів, які відсутні в стандарті EIA/TIA-568A.

До появи стандарту EIA/TIA велику роль відігравав американський стандарт системи категорій кабелів Underwriters Labs, розроблений спільно з компанією Anixter. Пізніше цей стандарт увійшов до стандарту EIA/TIA-568.

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

При стандартизації кабелів прийнято незалежний підхід. Це означає, що в стандарті обмовляються електричні, оптичні і механічні характеристики, яким повинен задовольняти той або інший тип кабелю або сполучного виробу – роз'єму, кросової коробки і т. п. Проте для якого протоколу призначений даний кабель, стандарт не обумовлює. Тому не можна придбати кабель для протоколу Ethernet або FDDI, потрібно просто знати, які типи стандартних кабелів підтримують протоколи Ethernet і FDDI.

У ранніх версіях стандартів визначалися тільки характеристики кабелів, без з'єднувачів. В останніх версіях стандартів

з'явилися вимоги до сполучних елементів (документи TSB-36 і TSB-40A, що увійшли потім у стандарт 568A), а також до ліній (каналів), що представляють типову збірку елементів кабельної системи, яка складається зі шнура від робочої станції до розетки, самої розетки, основного кабелю (завдовжки до 90 м для витої пари), точки переходу (наприклад, ще однієї розетки або жорсткого кросового з'єднання) і шнура до активного устаткування, наприклад концентратора або комутатора.

У стандартах кабелів враховуються досить багато характеристик, з яких найбільш важливі перераховані нижче.

Загасання (Attenuation). Загасання визначається як відносне зменшення амплітуди або потужності сигналу при переданні по лінії сигналу певної частоти. Загасання вимірюється в децибелах на метр для певної частоти або діапазону частот сигналу.

Загасання A обчислюються таким чином:

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

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

Перехресні наведення на ближньому кінці (Near End Cross Talk – NEXT). Визначають перешкодостійкість кабелю до внутрішніх джерел перешкод, коли електромагнітне поле сигналу, що передається виходом передавача по одній парі провідників, наводить на іншу пару провідників сигнал перешкоди. Якщо до другої пари буде підключений приймач, то він може прийняти наведену внутрішню перешкоду за корисний сигнал.

Показник NEXT, виражений у децибелах, обчислюється за формулою:

$$\text{NEXT} = 10 \log_{10} \frac{P_{\text{вих}}}{P_{\text{нав}}},$$

де $P_{\text{вих}}$ – потужність вихідного сигналу;

$P_{\text{нав}}$ – потужність наведеного сигналу. Чим менше значення NEXT, тим краще кабель. Показник NEXT вимірюються в децибелах для певної частоти сигналу.

Імпеданс (хвильовий опір) – це повний (активний і реактивний) опір в електричному ланцюзі. Імпеданс вимірюється в омах і є відносно постійною величиною для кабельних систем. Наприклад, для коаксіальних кабелів, що використовуються в стандартах Ethernet, імпеданс кабелю повинен складати 50 Ом. Для неекранованої витої пари найбільш часто використовуються значення імпедансу – 100 і 120 Ом. В області високих частот (100–200 МГц) імпеданс залежить від частоти.

Активний опір – це опір постійного струму в електричному ланцюзі. На відміну від імпедансу, активний опір не залежить від частоти і зростає зі збільшенням довжини кабелю.

Ємкість – це властивість металевих провідників накопичувати енергію. Два електричні провідники в кабелі, розділені діелектриком, є конденсатором, здатним накопичувати заряд. Ємкість є небажаною величиною, тому слід прагнути до того, щоб вона була якомога менше (іноді застосовують термін «паразитна ємкість»). Високе значення ємкості в кабелі приводить до спотворення сигналу і обмежує смугу пропускання лінії.

Рівень зовнішнього електромагнітного випромінювання або електричний шум. Електричний шум – це небажана змінна напруга в провіднику. Електричний шум буває двох типів: фоновий та імпульсний.

Електричний шум можна також розділити на низько-, середньо- і високочастотний. Джерелами фонового електричного шуму в діапазоні до 150 КГц є лінії електропередачі, телефони і лампи денного світла; у діапазоні від 150 КГц до 20 МГц – комп'ютери, принтери, ксерокси; у діапазоні від 20 МГц до 1 ГГц – телевізійні і радіопередавачі, мікрохвильові печі.

Основними джерелами імпульсного електричного шуму є мотори, перемикачі і зварювальні агрегати. Електричний шум вимірюється в мілівольтах.

Діаметр або площа перетину провідника. Для мідних провідників досить вживаною є американська система AWG (*American Wire Gauge*), яка вводить деякі умовні типи провідників, наприклад 22AWG, 24AWG, 26AWG. Чим більше номер типу провідника, тим менше його діаметр. В обчислювальних мережах найбільш вживаними є типи провідників, наведені вище. У європейських і міжнародних стандартах діаметр провідника указується в міліметрах.

Наведений перелік характеристик далеко не повний, причому в ньому представлені тільки електромагнітні характеристики, і його потрібно доповнити механічними і конструктивними характеристиками, що визначають тип ізоляції, конструкцію роз'єму та ін.

Крім універсальних характеристик, таких, наприклад, як загасання, які застосовують для всіх типів кабелів, існують характеристики, які застосовують тільки до певного типу кабелю. Наприклад, параметр *крок скручування дротів* використовується тільки для характеристики витой пари, а параметр NEXT застосовують тільки до багатопарних кабелів на основі витой пари.

Основна увага в сучасних стандартах приділяється кабелям на основі витой пари і волоконно-оптичним кабелям.

Пропускна здатність (*throughput*) **каналу** характеризує максимально можливу швидкість передання даних по каналу зв'язку.

Далі вживатимемо канал і лінію зв'язку як синоніми. Пропускна здатність ліній зв'язку і комунікаційного мережевого устаткування традиційно вимірюється в бітах за секунду, а не в байтах за секунду. Це пов'язано з тим, що дані в мережах передаються послідовно, тобто побітно, а не паралельно, байтами, як це відбувається між пристроями усередині комп'ютера. Такі одиниці вимірювання, як кілобіт, мегабіт або гігабіт, у мережових технологіях строго відповідають ступеням 10 (тобто кілобіт – це 1000 біт, а мегабіт – це 1 000 000 біт), як це прийнято у всіх галузях науки і техніки, а не близьким до цих чисел ступеням 2, як це прийнято в програмуванні, де приставка «кіло» дорівнює $2^{10} = 1024$, а «мега» – $2^{20} = 1048576$.

До повітряних ліній мереж відносяться сталеві і кольорові (мідні і біметалічні) лінії.

Сталеві і кольорові лінії мереж прокладаються над поверхнею землі на спеціальних опорах. Для лінії мереж, як правило, використовуються неізолювані (голі) дроти. Для підвіски дротів на опорах волоконних мереж зв'язку застосовують крюки і траверси.

Повітряні лінії мереж бувають одно- і дводротяними. Однодротяні повітряні лінії мереж застосовуються досить рідко і використовуються лише для організації телеграфного зв'язку по схемі «дріт – земля».

Обидва дроти ланцюга повинні мати однакові параметри, наприклад, зроблені з однакового матеріалу, одного діаметру, підвішуватися на ізоляторах однієї марки. Для підвищення захищеності ланцюга від зовнішніх перешкод і впливів сусідніх ланцюгів дроту ланцюги схрещують за відповідною схемою.

Кабель є електропровідником системи, що складається з ізольованих провідників, поміщених у загальну ізоляційну і захисну оболонку.

До кабельних ліній відносять:

- симетричні кабелі;
- коаксіальні кабелі.

8.2. СИМЕТРИЧНІ КАБЕЛІ

Симетричний кабельний ланцюг складається з двох максимально однакових в електричному і конструктивному відношенні ізольованих провідників.

- За спектром частот, що передаються, кабелі розділяються на:
- низькочастотні симетричні (до 10 кГц);

- високочастотні симетричні (10 кГц – 10 МГц);
- коаксіальні.

Струмopровідні жили (звичайно круглої форми) кабелів зв'язку повинні володіти високою електричною провідністю, гнучкістю і достатньою механічною міцністю. Найбільш поширеними матеріалами для виготовлення кабельних жил є мідь і алюміній. Діаметр мідних жил вибирається з ряду 0,5; 0,6; 0,8; 0,9; 1,2; 1,4. Для отримання алюмінієвої жили, яка еквівалентна мідній за опором, її діаметр повинен бути збільшений у 1,3 рази.

Для ізоляції жил дrotів застосовують папір, стирофлекс, поліетилен, полівінілхлорид та інші ізолятори. Ізольовані жили скручують або парами, або по чотири – зоряне скручування. Необхідну кількість груп об'єднують у загальнокабельне скручування – звичайно шарами. Число пар 5, 10, 20 ... 1200.

Як матеріал для оболонок використовують свинець, алюміній, поліетилен, полівінілхлорид та ін. Кабелі, призначені для ґрунту або води, забезпечують сталеву броню зі сталевих стрічок (дроту) і пряжі, просоченої смолою.

Кожному кабелю присвоюється марка з декількох букв і цифр, у якій зашифрована галузі застосування, умови прокладки, спектр частот, що передаються, конструкція, матеріал і форма ізоляції, система скручування провідників, матеріал і конструкція захисних покривів.

Кабелі на основі неекранованої витой пари

Мідний неекранований кабель UTP залежно від електричних і механічних характеристик розділяється на 5 категорій (Category 1 – Category 5).

Кабелі *категорій 1 і 2* були визначені в стандарті EIA/TIA-568, але до стандарту 568A вже не увійшли, як застарілі.

Кабелі *категорії 1* застосовуються там, де вимоги до швидкості передання мінімальні. Зазвичай це кабель для цифрового і аналогового передання голосу і низькошвидкісного (до 20 Кбіт/с) передання даних. До 1983 р. це був основний тип кабелю для телефонної розводки.

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

Кабелі *категорії 3* були стандартизовані в 1991 р., коли був розроблений Стандарт телекомунікаційних кабельних систем для комерційних будівель (EIA-568), на основі якого потім був створений діючий стандарт EIA-568A. Стандарт EIA-568 визначив електричні характеристики кабелів категорії 3 для

частот у діапазоні до 16 МГц, що підтримують таким чином високошвидкісні мережеві додатки. Кабель категорії 3 призначений як для передавання даних, так і для передавання голосу. Крок скручування дротів рівний приблизно 3 витки на 1 фут (30,5 см). Кабелі категорії 3 зараз складають основу багатьох кабельних систем будівель, у яких вони використовуються для передавання і голосу, і даних.

Кабелі *категорії 4* є дещо ліпшим варіантом ніж кабелі категорії 3. Кабелі категорії 4 зобов'язані витримувати тести на частоті передавання сигналу 20 МГц і забезпечувати підвищену перешкодостійкість та низькі втрати сигналу. Кабелі цієї категорії добре підходять для застосування в системах зі збільшеними відстанями (до 135 м) і в мережах Token Ring з пропускну здатністю 16 Мбіт/с. На практиці використовуються рідко.

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

Найбільш важливі електромагнітні характеристики кабелю категорії 5 мають наступні значення:

- повний хвильовий опір у діапазоні частот до 100 МГц рівно 100 Ом (стандарт ISO 11801 допускає також кабель із хвильовим опором 120 Ом);

- величина перехресних наведень NEXT залежно від частоти сигналу повинна приймати значення не менше 74 дБ на частоті 150 кГц і не менше 32 дБ на частоті 100 МГц.

Ланцюги ліній зв'язку постійно знаходяться під впливом сторонніх електромагнітних полів різного походження. Розрізняють дві основні групи джерел сторонніх полів:

- 1) внутрішні – сусідні фізичні і штучні ланцюги даної лінії зв'язку;

- 2) зовнішні – енергетично і конструктивно не пов'язані з лінією зв'язку.

Зовнішні джерела перешкод, у свою чергу, за своїм походженням діляться на:

- природні – грозові розряди, сонячна радіація та ін.;

– створені людиною – високовольтні лінії передачі, радіостанції, лінії електрифікованих залізниць, електричні мережі промислових підприємств і окремі енергоємні пристрої.

Сторонні електромагнітні поля індукують у ланцюгах ліній зв'язку перешкоди, які не тільки знижують якість передання, але іноді порушують великі напруги і струми, що приводять до руйнування ліній зв'язку і апаратури. Вказані дії називають *електромагнітними впливами*, або *впливами на ланцюги ліній зв'язку*.

Дана проблема є спільною для всіх систем і пристроїв телекомунікацій і називається *проблемою електромагнітної сумісності*. Суть її полягає в тому, що в процесі проектування, будівництва і експлуатації телекомунікаційних пристроїв і систем необхідно враховувати дві суперечливі вимоги:

1) необхідно забезпечити достатній для нормальної роботи телекомунікаційних систем захист від дії на них сторонніх електромагнітних полів;

2) необхідно обмежити допустимими значеннями рівні впливу електромагнітних полів проєктованих пристроїв і систем на інші пристрої.

При кількісній оцінці рівня взаємних впливів звичайно розглядають два ланцюги: що впливає (створює електромагнітне поле) і схильний до впливу (у якому індукуються перешкоди) (див. рис. 8.1).

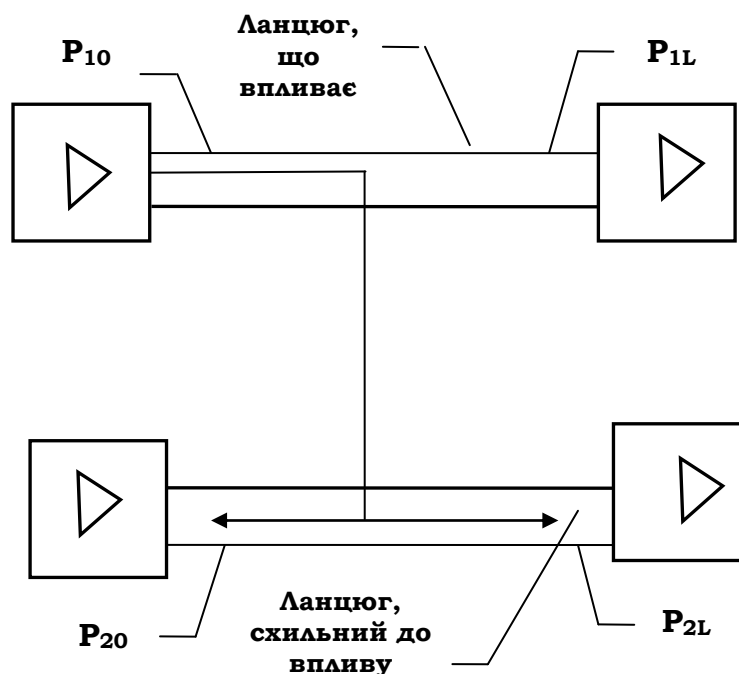


Рис. 8.1. Взаємний вплив ланцюгів

Ближнім кінцем лінії називають той, до якого підключений генератор, дальнім кінцем – той, до якого підключене навантаження ланцюга. Відповідно розглядаються потужності сигналів

у ланцюгах: P_{10} – на ближньому кінці ланцюга, що впливає; P_{1L} – на дальньому кінці ланцюга, що впливає; P_{20} – на ближньому кінці ланцюга, схильного до впливу; P_{2L} – на дальньому кінці ланцюга, схильного до впливу.

Кількісно захищеність від перехідних перешкод через взаємні електромагнітні впливи оцінюється за кількома показниками, зокрема перехідним загасанням на ближньому кінці лінії (*near end cross talk* – NEXT), перехідним загасанням на дальньому кінці лінії (*far end cross talk* – FEXT).

Загасання має граничні значення від 0,8 дБ (на частоті 64 кГц) до 22 дБ (на частоті 100 МГц).

Коефіцієнт загасання α , дБ/км. Залежить від властивостей матеріалів провідників і ізоляційного матеріалу. Якнайкращими властивостями (малим опором) володіють мідь і срібло. Коефіцієнт загасання залежить також від геометричних розмірів провідників. СК (симетричні кабелі) з великими діаметрами провідників володіють меншим коефіцієнтом загасання. Коефіцієнт загасання КК (коаксіального кабелю) залежить від співвідношення діаметрів зовнішнього і внутрішнього провідника (див. рис. 8.2). Оптимальними співвідношеннями є (матеріал зовнішнього провідника): для міді – 3,6, для алюмінію – 3,9, для свинцю – 5,2.

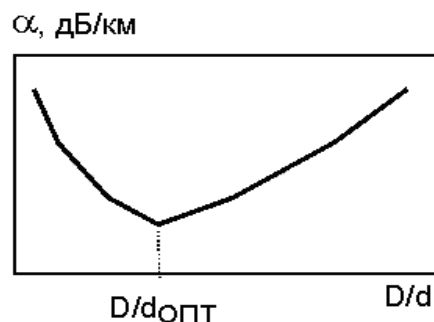


Рис. 8.2. Залежність коефіцієнта загасання КК від співвідношення діаметрів провідників: α – коефіцієнт загасання КК; D , d – діаметри провідників; $D/d_{\text{опт}}$ – точка оптимального співвідношення

Дуже важливою характеристикою, що фактично визначає широкосмугову систему зв'язку, є залежність коефіцієнта загасання від частоти (див. рис. 8.3). Якщо визначений граничний коефіцієнт загасання $\alpha_{\text{ГР}}$ (звичайно він визначається можливостями підсилювачів або регенераторів), то даному коефіцієнту відповідає гранична частота пропускання системи $f_{\text{ГР}}$. Смуга пропускання системи не перевищує граничної частоти пропускання.

Швидкість поширення v , км/мс. Частотна залежність швидкості поширення показана на рис. 8.4. Із зростанням частоти швидкість поширення збільшується, наближаючись

до швидкості світла у вакуумі ($V_c = 300$) км/мс. Даний параметр залежить також від властивостей діелектрика, вживаного в кабелі.

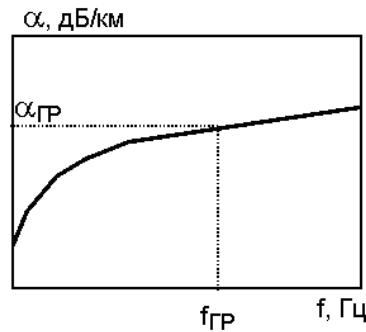


Рис. 8.3. Частотна залежність коефіцієнта загасання металевого кабелю: α – коефіцієнт загасання металевого кабелю; $\alpha_{ГР}$ – граничний коефіцієнт загасання металевого кабелю; f – частота

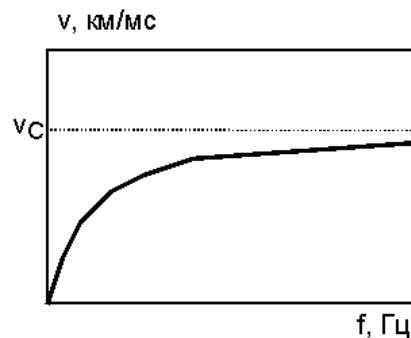


Рис. 8.4. Частотна залежність швидкості поширення електромагнітної хвилі: v_c – швидкість поширення електромагнітної хвилі, f – частота

Активний опір не повинен перевищувати 9,4 Ом на 100 м; місткість кабелю не повинна перевищувати 5,6 нф на 100 м. Усі кабелі UTP, незалежно від їх категорії, випускаються в 4-парному виконанні. Кожна з чотирьох пар кабелю має певний колір і крок скручування. Звичайно дві пари призначені для передавання даних, а дві – для передавання голосу.

Для з'єднання кабелів з устаткуванням використовуються вики і розетки RJ-45, що представляють 8-контактні роз'єми, схожі на звичайні телефонні роз'єми RJ-11.

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

Кабелі на основі екранованої витой пари

Екранована вита пара STP добре захищає сигнали, що передаються від зовнішніх перешкод, а також менше випромінює електромагнітних коливань зовні, що захищає, у свою чергу, користувачів мереж від шкідливого для здоров'я випромінювання. Наявність екрана, що заземляється, підвищує вартість кабелю і ускладнює його прокладку, оскільки вимагає виконання якісного заземлення. Екранований кабель застосовується тільки для передавання даних.

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

Основним типом екранованого кабелю є кабель Type 1 стандарту IBM. Він складається з 2-х пар скручених дротів, екранованих дротами, які заземляються. Електричні параметри кабелю Type 1 приблизно відповідають параметрам кабелю UTP категорії 5. Проте хвильовий опір кабелю Type 1 – 150 Ом (UTP категорії 5 має хвильовий опір 100 Ом), тому просте «поліпшення» кабельної проводки мережі шляхом заміни неекранованої пари UTP на STP Type 1 неможливе. Трансивери, розраховані на роботу з кабелем, що має хвильовий опір 100 Ом, погано працюватимуть на хвильовий опір 150 Ом. Тому при використанні STP Type 1 необхідні відповідні трансивери. Такі трансивери є в мережевих адаптерах Token Ring, оскільки ці мережі розроблялися для роботи на екранованій витій парі. Деякі інші стандарти також підтримують кабель STP Type 1, наприклад, 100VG-AnyLAN, а також Fast Ethernet (хоча основним типом кабелю для Fast Ethernet є UTP категорії 5). У випадку якщо технологія може використовувати UTP і STP, потрібно переконатися, на який тип кабелю розраховані трансивери. Кабель STP Type 1 включений у стандарти EIA/TIA-568A, ISO 11801 і EN 50173, тобто набув міжнародного статусу. Екрановані виті пари використовуються також у кабелі IBM Type 2, який представляє кабель Type 1 з доданими 2 парами неекранованого дроту для передавання голосу.

Для приєднання екранованих кабелів до устаткування використовуються роз'єми конструкції IBM.

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

8.3. КОАКСІАЛЬНИЙ КАБЕЛЬ

Коаксіальний ланцюг складається з двох циліндрів із суміщеною віссю, причому суцільний внутрішній циліндровий

провідник розташований по осі усередині порожнистого циліндра (див. рис. 8.5).

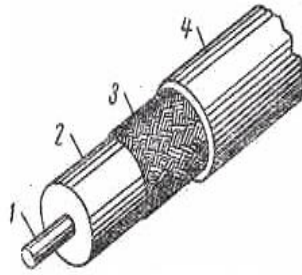


Рис. 8.5. Конструкція гнучкого коаксіального кабелю:
1 – внутрішній дріт, 2 – ізоляція,
3 – зовнішній дріт, 4 – захисна оболонка

При переданні електричних сигналів по коаксіальному кабелю (КК) струми провідності циркулюють по центральному провіднику і оболонці, що породжує електромагнітну хвилю. Взаємодія електричних полів внутрішнього і зовнішнього провідників КК така, що його зовнішнє поле рівне нулю. Силові лінії магнітного поля, що створюються зустрічними струмами в КК, розташовуються у вигляді концентричних кіл усередині оболонки, взаємознищуючись зовні.

Електричне поле також замикається усередині КК по радіальних напрямках між центральним провідником і оболонкою, і тому поза кабелем поле рівне нулю. Таким чином, у КК через відсутність зовнішнього поля немає перешкод на сусідні ланцюги, і крім того, втрат в оточуючих його масах, тому вся енергія поширюється усередині кабелю.

Існує велика кількість типів коаксіальних кабелів, що використовуються у мережах різного типу (див. рис. 8.6) – телефонних, телевізійних і комп'ютерних.

Основні типи і характеристики КК

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

RG-58/U, RG-58 A/U і RG-58 C/U – різновиди «тонкого» коаксіального кабелю для мереж Ethernet 10Base-2. Кабель RG-58/U має суцільний внутрішній провідник, а кабель RG-58 A/U – багатожильний. Усі ці різновиди кабелю мають хвильовий опір 50 Ом, але володіють гіршими механічними і електричними характеристиками в порівнянні з «товстим» коаксіальним кабелем: тонкий внутрішній провідник має набагато

більшу гнучкість, зручну при монтажі. Загасання в цьому типі кабелю вище, ніж в «товстому» коаксіальному кабелі, що призводить до необхідності зменшувати довжину кабелю для отримання однакового загасання в сегменті. Для з'єднання кабелів з устаткуванням використовується роз'єм типу BNC.

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

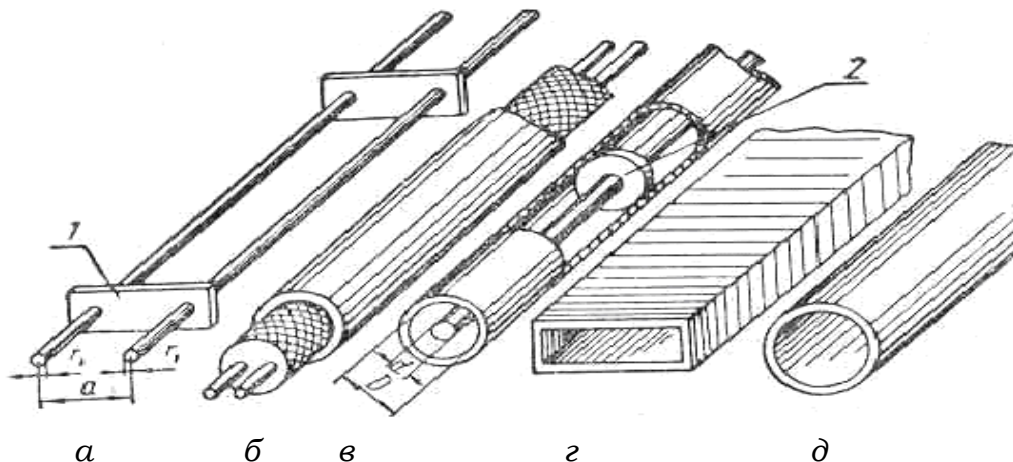


Рис. 8.6. Лінії передачі:

де: *a* – відкрита дводротова лінія; *б* – екранована дводротова лінія;
в – коаксіальна лінія; *г* – прямокутний хвилевід; *д* – круглий хвилевід

RG-62 – кабель із хвильовим опором 93 Ом, використовувався в мережах ArcNet, устаткування яких сьогодні практично не випускається.

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

Хвилеводи і світловоди (див. рис 8.6 *г, д*). Хвилевод – канал для розповсюдження хвиль. Конструктивно хвилевод є порожнистою металевою трубою круглого або прямокутного перетину, виготовленою з добре провідного матеріалу. Відомі такі хвилеводи інших конструкцій – еліптичні, П-образні, Н-образні. Звичайно хвилевод робляться біметалічними – зовнішня силова частина виготовляється з конструкційного матеріалу – сталь, сплави, а внутрішня – із матеріалів з високою провідністю – мідь, срібло, золото.

Передавання по хвилеводу електромагнітної енергії може бути описане з достатньою точністю тими ж законами, що й при вільному випромінюванні електромагнітних хвиль у просторі.

Теоретично і експериментально встановлено, що при передаванні по круговому хвилеводу діаметром 0,6 см може бути досягнута дальність близько 40 км.

У хвилеводі електромагнітне поле повністю екранується і втрат на випромінювання не повинно бути. Діелектричні втрати у хвилеводі також дуже малі. Втрати в металі у хвилевода істотно менше, ніж у симетричного або коаксіального ланцюга. У коаксіального кабелю, наприклад, більша частина втрат (80 %) викликається внутрішнім дротом, оскільки периметр його перетину набагато менший, ніж зовнішнього. Таким чином, мінімальні втрати у хвилевода, потім – у коаксіального кабелю, і, нарешті, у симетричного (максимальні). Цінністю хвилеводу є також те, що по ньому можна пропускати більшу потужність, ніж по коаксіальному кабелю.

Резюмуючи, можна відзначити, що *достоїнствами хвилеводу є:*

- можливість передавання вельми високих частот – близько 1011–1012 Гц;
- повне екранування поля;
- малі втрати при великій потужності;
- простота конструкції.

До недоліків хвилеводів відносяться:

– наявність критичної частоти, у зв'язку, з чим хвилевод не пропускає частот, довжини хвиль яких більші, ніж діаметр хвилеводу;

– жорсткі вимоги до точності виготовлення, прокладки і монтажу.

Так, допустимий вигін на довжині 10 м не повинен перевищувати 5 см.

Передавання електромагнітної енергії по хвилеводу можливе при заміні металеві труби на діелектричну. У подібному хвилеводі струми провідності відсутні і їх замінюють струми зсуву. Діелектричний хвилевод може бути або порожнистим, або монолітним стрижнем, виконаним із високочастотного діелектрика з діелектричною проникністю $S > 1$.

Встановлено, що діелектричний хвилевод, як і металевий, має критичну частоту, починаючи з якої хвилевод діє як направляюча система передавання.

Діелектричні хвилеводи в порівнянні з металевими мають менші втрати і, відповідно, загасання, проте вони як відкриті системи істотно схильні до дії перешкод і самі випромінюють енергію, створюючи перешкоди в сусідніх ланцюгах.

У даний час діелектричні хвилеводи застосовуються в основному для передавання сигналів на короткі відстані (десятки метрів) у діапазоні частот 10–00 ГГц. Мінімальна величина загасання при цьому досягається близько 20 дБ/км (при частоті 50 ГГц і перетині діелектричного хвилеводу $0,8 \times 2,5$ мм).

8.4. ВОЛОКОННО-ОПТИЧНІ ЛІНІЇ ЗВ'ЯЗКУ

Волоконно-оптичні лінії зв'язку (ВОЛЗ) мають низку істотних переваг у порівнянні з лініями зв'язку на основі металевих кабелів. До них відносяться: велика пропускна здатність, мале загасання, малі маса і габарити, висока перешкодостійкість, надійна техніка безпеки, практично відсутні взаємні впливи, мала вартість через відсутність у конструкції кольорових металів.

У ВОЛЗ застосовують електромагнітні хвилі оптичного діапазону. Нагадаємо, що видиме оптичне випромінювання знаходиться в діапазоні довжин хвиль 380–760 нм. Практичне застосування у ВОЛЗ одержав інфрачервоний діапазон, тобто випромінювання з довжиною хвилі більше 760 нм.

Принцип поширення оптичного випромінювання уздовж оптичного волокна (ОВ) заснований на віддзеркаленні від межі середовищ із різними показниками заломлення (див. рис. 8.7). Оптичне волокно виготовляється з кварцового скла у вигляді циліндрів із суміщеними осями і різними коефіцієнтами заломлення. Внутрішній циліндр називається серцевиною ОВ, а зовнішній шар – оболонкою ОВ.

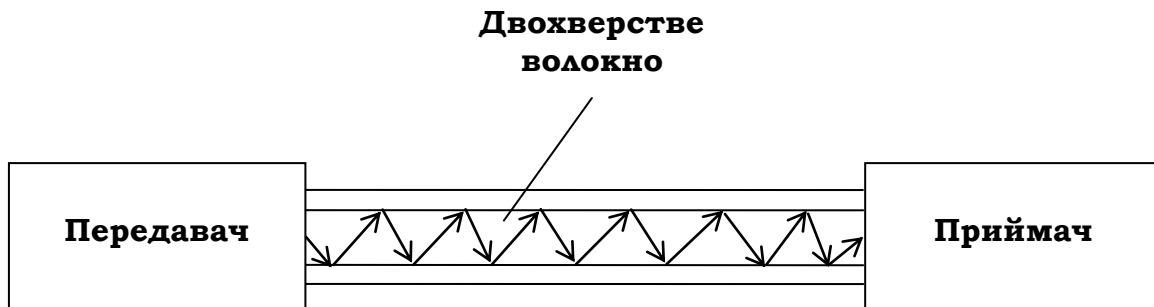


Рис. 8.7. Принцип поширення оптичного випромінювання

Кут повного внутрішнього віддзеркалення, при якому падаюче на межу двох середовищ випромінювання повністю відбивається без проникнення в зовнішнє середовище, визначається співвідношенням, де n_1 – показник заломлення сердечника ОВ, n_2 – показник заломлення оболонки ОВ, причому $n_1 > n_2$.

Залежно від виду профілю показника заломлення сердцевини розрізняють ступінчасті і градієнтні ОВ. У ступінчастих ОВ показник заломлення сердцевини постійний (див. рис. 8.8 а). У градієнтних ОВ показник заломлення сердцевини плавно міняється уздовж радіусу від максимального значення на осі до значення показника заломлення оболонки (див. рис. 8.8 б).

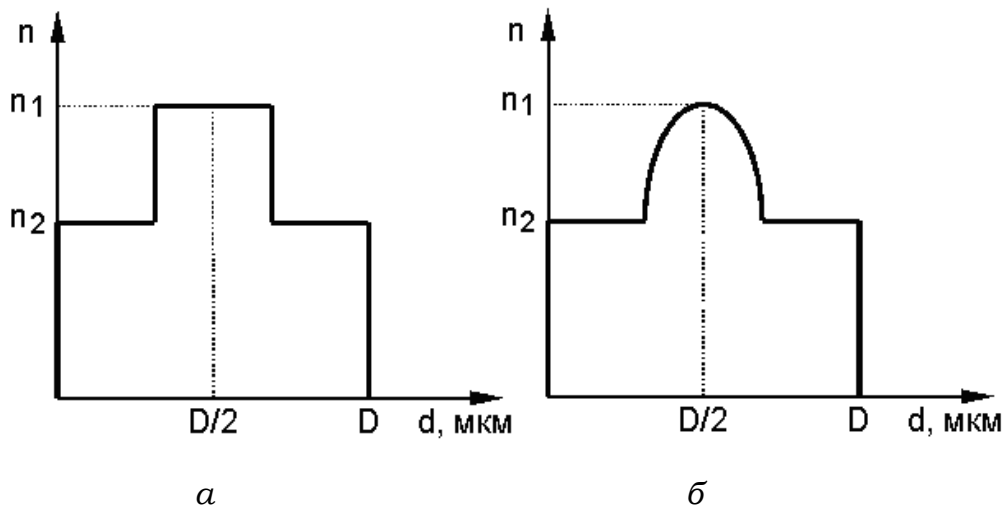


Рис. 8.8. Профіль показника заломлення ступінчастого (а) і градієнтного (б) ОВ

Залежно від розподілу показника заломлення і від величини діаметра сердечника розрізняють:

- багатомодове волокно із ступінчастою зміною показника заломлення (див. рис. 8.9 а);
- багатомодове волокно з плавною зміною показника заломлення (див. рис. 8.9 б);
- одномодове волокно (див. рис. 8.9 в).

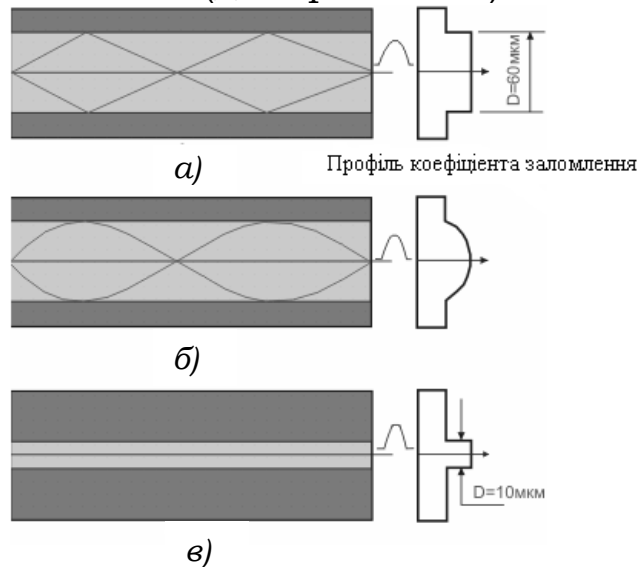


Рис. 8.9. Типи оптичного кабелю

Поняття «мода» описує режим поширення світлових променів у внутрішньому сердечнику кабелю. В одномодовому кабелі (*Single Mode fiber – SMF*) використовується центральний провідник дуже малого діаметра, який можна порівняти з довжиною хвилі світла від 5 до 10 мкм. При цьому практично всі промені світла поширюються уздовж оптичної осі світловоду, не відображаючись від зовнішнього провідника. Смуга пропускання одномодового кабелю дуже широка – до сотень гігагерц на

кілометр. Виготовлення тонких якісних волокон для одномодового кабелю представляє складний технологічний процес, що робить одномодовий кабель достатньо дорогим. Коротше за траєкторією – швидше поширюється. Крім того, у волокно такого маленького діаметра достатньо складно направити пучок світла, не втративши при цьому значну частину енергії.

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

У багатомодових кабелях у внутрішньому провіднику одночасно існує кілька світлових променів, що відображаються від зовнішнього провідника під різними кутами. Кут віддзеркалення променя називається модою променя. У багатомодових кабелях із плавною зміною коефіцієнта заломлення режим поширення кожної моди має складніший характер.

Багатомодові кабелі мають вузьку смугу пропускання – від 500 до 800 МГц/км. Звуження смуги відбувається через втрати світлової енергії при віддзеркаленнях, а також через інтерференції променів різних мод. На рисунку 8.10 наведений приклад конструкції оптичного кабелю.

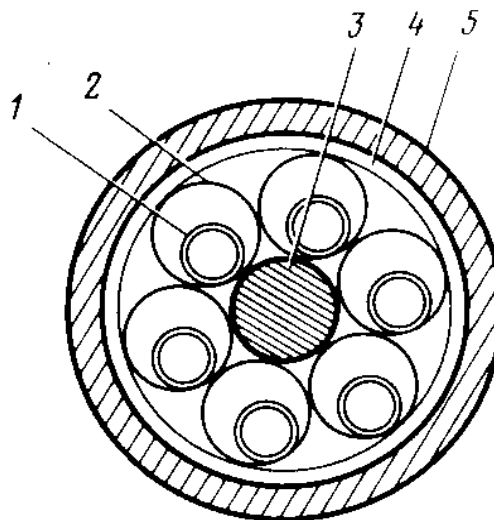


Рис. 8.10. Конструкція оптичного кабелю:

1 – ОВ, 2 – поліетиленова трубка, 3 – силовий елемент,
4 і 5 – відповідно внутрішня і зовнішня поліетиленові оболонки

Загасання ОВ неоднорідне для різних довжин хвиль. Залежність коефіцієнта загасання ОВ від робочої довжини хвилі наведена на рисунку 8.11. Дана залежність має три мінімуми, звані вікнами прозорості. Історично першим було освоєне перше вікно прозорості на робочій довжині хвилі 0,85 мкм.

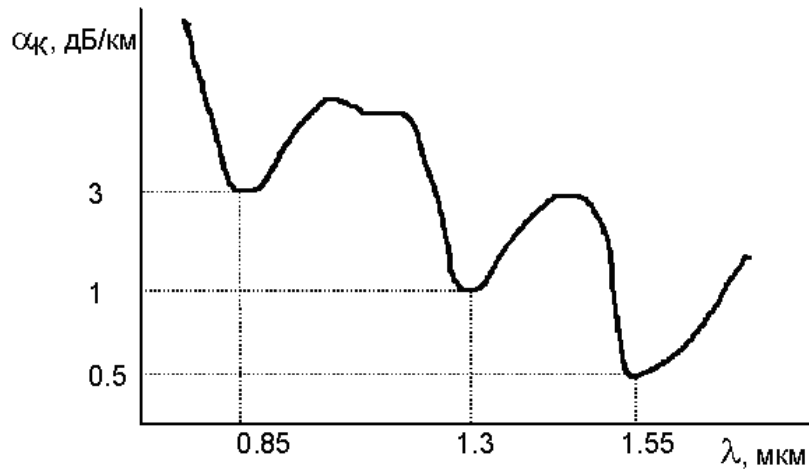


Рис. 8.11. Спектральна характеристика коефіцієнта загасання ОВ

Перші напівпровідникові випромінювачі (лазери і світлодіоди) і фотоприймачі були розроблені саме для даної довжини хвилі. Коефіцієнт загасання в першому вікні досить значний і складає одиниці дБ/км. Пізніше були створені випромінювачі і фотоприймачі, здатні працювати на великих довжинах хвиль (1,3 і 1,55 мкм). Сучасні системи зв'язку звичайно використовують друге або третє вікно з малими коефіцієнтами загасання. Сучасна технологія дозволяє одержати ОВ з коефіцієнтом загасання порядку сотих долей дБ/км.

Як джерела випромінювання світла у волоконно-оптичних кабелях застосовуються:

- світлодіоди;
- напівпровідникові лазери.

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

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

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

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

Волоконно-оптичні кабелі приєднують до устаткування роз'ємами MIC, ST і SC.

Волоконно-оптичні кабелі володіють відмінними характеристиками всіх типів: електромагнітними, механічними (добре гнуться, а у відповідній ізоляції володіють хорошою механічною міцністю). Проте в них є один серйозний недолік – складність з'єднання волокон із роз'ємами і між собою при необхідності нарощування довжини кабелю.

Сама вартість волоконно-оптичних кабелів ненабагато перевищує вартість кабелів на витій парі, проте проведення монтажних робіт з оптоволоком обходиться набагато дорожче через трудомісткість операцій і високу вартість вживаного монтажного устаткування. Так, приєднання оптичного волокна до роз'єму вимагає проведення високоточного обрізання волокна в площині строго перпендикулярній осі волокна, а також виконання з'єднання шляхом складної операції склеювання, а не обжимання, як це робиться для витієї пари. Виконання ж неякісних з'єднань відразу різко зводить до нуля пропускання волоконно-оптичних кабелів і ліній.

8.5. ПРИСТРОЇ ОБМІНУ ДАНИМИ

Модем – пристрій, призначений для обміну інформацією між віддаленими комп'ютерами по каналах зв'язку (Модулятор + Демодулятор).

При цьому під каналом зв'язку розуміють фізичні лінії (дротяні, оптоволоконні, кабельні, радіочастотні), спосіб їх використання (комутовані і виділені) і спосіб передавання даних (цифрові і аналогові сигнали). Залежно від типу каналу зв'язку пристрої приймання-передавання підрозділяють на радіомодеми, кабельні модеми та ін. Найбільш широке застосування знайшли модеми, орієнтовані на підключення до комутованих телефонних каналів зв'язку.

Цифрові дані, що надходять у модем із комп'ютера, перетворюються в ньому шляхом модуляції (по амплітуді, частоті, фазі) відповідно до вибраного стандарту (протоколом) і прямують у телефонну лінію. Модем-приймач, що розуміє даний протокол, здійснює зворотне перетворення (демодуляцію) і пересилає відновлені цифрові дані у свій комп'ютер. Таким чином,

забезпечується видалений зв'язок між комп'ютерами і обмін даними між ними.

До основних споживчих параметрів модемів відносяться:

1. Продуктивність (біт/с);
2. Підтримувані протоколи зв'язку і корекції помилок;
3. Шинний інтерфейс, якщо модем внутрішній.

Від продуктивності модема залежить об'єм даних, які можна передати в одиницю часу. Від підтримуваних протоколів залежить ефективність взаємодії даного модема з суміжними модемами (вірогідність того, що вони вступають у взаємодію один з одним при оптимальних налаштуваннях). Від шинного інтерфейса в даний час поки залежить тільки простота встановлення і налаштування модема (надалі при загальному вдосконаленні каналів зв'язку шинний інтерфейс почне впливати і на продуктивність).

Також важливими показниками в сучасних модемах є наявність режиму корекції помилок і режиму стиснення даних. Перший режим забезпечує додаткові сигнали, за допомогою яких модеми здійснюють перевірку даних на двох кінцях лінії і відкидають немарковану інформацію, а другий стискає інформацію для швидшого і чіткішого її передання, а потім відновлює її на одержуючому модемі. Обидва ці режими помітно збільшують швидкість і чистоту передання інформації.

Одна з передових фірм-виробників модемів Hayes Microcomputer Products прийняла основні стандарти для команд модемів, включаючи набір AT-команд, за допомогою яких користувач може безпосередньо управляти роботою модема. Сьогодні Hayes-стандартами користуються переважна більшість фірм у всьому світі, і кращі модеми є Hayes-сумісними.

Також існують світові стандарти швидкості модема стиснення даних і корекції помилок. Ці стандарти встановлюються комітетом ITU-T (стандарт CCITT) і фірмою Microsoft (стандарт MNP). Найкращі модеми відповідають обом цим стандартам.

Найпоширеніші стандарти CCITT:

1. Стандарт швидкості 9600 bps – V.32 і швидкості 14400 bps – V.32 bis;
2. Стандарт корекції помилок – V.42;
3. Стандарт стиснення даних із коефіцієнтом 4 : 1 – V.42 – V.42 bis.

Фірма ZyXEL і US Robotics проводять найшвидкісніші і найякісніші модеми і факс-модеми. Дуже дорогі суперсучасні модеми ZyXEL мають можливість відтворення голосу, записаного в цифровому режимі, і стиснення мовних сигналів, що дозволяє використовувати їх як автовідповідачі. Також деякі

моделі ZeXEL U-1496 і US Robotics Courier забезпечені перемикачем (мова/дані), вбудованим тестуванням і іншими корисними функціями.

Серед новинок останніх років у світі модемів можна також виділити модеми для Notebook, що поставляються на платі типу РСМСІА. Ця плата дуже зручна своєю компактністю, вони дозволяють комп'ютеру не віддавати вільний СОМ-порт під зовнішній модем, але все ж таки вони набагато дорожче, ніж звичайний модем.

Інтерфейси

Ви можете використовувати модем у вашому комп'ютері за допомогою одного з двох інтерфейсів. Ними є:

1. MN-5 послідовний інтерфейс RS-232.
2. MNp-5 чотирьохконтактний телефонний кабель RJ-11.

Наприклад, зовнішній модем підключається до комп'ютера за допомогою кабелю RS-232, а до телефонної лінії – за допомогою кабелю RJ-11.

Флеш-пам'ять вашого модема

Флеш-пам'ять – постійна пам'ять або ПЗП, яка може бути стерта і знов запрограмована.

Перепрограмуванню підлягають усі модеми, у назві яких присутній рядок *V.Everything*. Крім того, модеми *Courier V.34 dual standart* підлягають програмній модернізації у випадку, якщо в рядку Options відповідає на команду *AT17* присутній протокол *V/FC*. Якщо ж в модемі немає цього протоколу, то модернізація в *Courier V.Everything* проводиться заміною дочірньої плати.

Існують дві модифікації модемів *Courier V.Everything* – з так званою частотою супервізора 20.16 МГц і 25 МГц. Для кожного з них існують свої версії прошивок, і вони не є взаємозамінними, тобто прошивка від моделі 20.16 МГц не підійде для моделі 25 МГц, і навпаки.

8.6. ПЛАТИ МЕРЕЖЕВОГО АДАПТЕРА

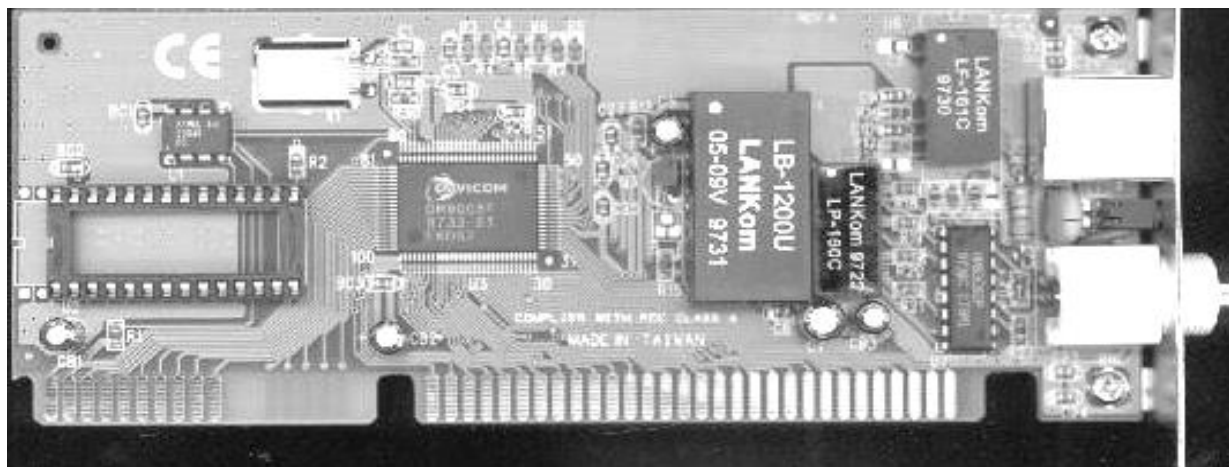
Плата мережевого адаптера (див. рис. 8.12) виступає як фізичний інтерфейс, або з'єднання, між комп'ютером і мережевим кабелем. Плата вставляється в слоти розширення всіх мережевих комп'ютерів і серверів.

Щоб забезпечити фізичне з'єднання між комп'ютером і мережею, до плати (після її установки) підключається мережевий кабель.

Призначення плати мережевого адаптера:

– підготовка даних, що надходять від комп'ютера, до передавання по мережевому кабелю;

- передавання даних іншому комп'ютеру;
- управління потоком даних між комп'ютером і кабельною системою.



Плата мережевого адаптера, крім того, приймає дані з кабелю і переводить їх у форму, зрозумілу центральному процесору комп'ютера.

Плата мережевого адаптера складається з апаратної частини і вбудованих програм, записаних у ПЗП. Ці програми реалізують функції підрівнів *Управління* логічним зв'язком і *Управління* доступом до середовища *Канального* рівня моделі OSI.

Підготовка даних

Перш ніж надіслати дані в мережу плата мережевого адаптера повинна перевести їх із форми, зрозумілої комп'ютеру, у форму, у якій вони можуть передаватися по мережевому кабелю.

Усередині комп'ютера дані передаються по шинах. Як правило, це декілька провідників, розташованих близько один до одного. Оскільки ліній декілька, то і біти даних можуть передаватися по ним групами, а не послідовно.

Шини, які використовувалися в перших персональних комп'ютерах IBM, були відомі як 8-розрядні шини: вони могли передавати групи по 8 бітів даних. IBM PC/AT® має 16-розрядну шину; це означає, що вона здатна передавати відразу 16 бітів. Багато сучасних комп'ютерів оснащені вже 32-розрядною шиною. Часто говорять, що дані по шині комп'ютера передаються паралельно (*parallel*), оскільки 16 бітів або 32 біти рухаються паралельно один одному. Уявіть, що 16-розрядна шина – це 16-смугова автострада, по якій поряд (паралельно) рухаються 16 машин, кожна з яких перевозить один біт.

У мережевому кабелі дані повинні переміщатися у вигляді потоку бітів. При цьому говорять, що відбувається послідовне передавання, тому що біти слідуєть один за одним. Іншими

словами, кабель – це дорога з однією смутою. По таких «дорогах» дані в кожен момент часу рухаються в одному напрямі.

Плата мережевого адаптера приймає паралельні дані і організовує їх для послідовного (*serial*) побітового передання. Цей процес завершується переведенням цифрових даних комп'ютера в електричні і оптичні сигнали, які і передаються по мережевих кабелях. Відповідає за це перетворення трансивер.

Мережева адреса

Крім перетворення даних, плата мережевого адаптера повинна вказати своє місцезнаходження, або адресу, – щоб її могли відрізнити від решти плат.

Мережеві адреси (*network address*) визначені комітетом IEEE. Цей комітет закріплює за кожним виробником плати мережевого адаптера деякий інтервал адрес. Виробники «зашивають» ці адреси в мікросхеми. Завдяки цьому кожна плата, а отже, й кожен комп'ютер мають унікальну адресу в мережі.

При прийомі даних від комп'ютера і підготовці їх до передання по мережевому кабелю плата мережевого адаптера бере участь також в інших операціях.

1. Комп'ютер і плата мережевого адаптера повинні бути пов'язані один з одним, щоб здійснювати передання даних (від комп'ютера до плати). Якщо плата може використовувати прямий доступ до пам'яті, комп'ютер виділяє їй деяку область своєї пам'яті.

2. Плата мережевого адаптера запрошує у комп'ютера дані.

3. Шина комп'ютера передає дані з його пам'яті плати мережевого адаптера.

Часто дані поступають швидше, ніж їх здатна передати плата мережевого адаптера, тому тимчасово вони поміщаються в буфер.

Передання і управління даними

Перш ніж надіслати дані по мережі плата мережевого адаптера проводить електронний діалог із приймаючою платою, під час якого вони «обговорюють»:

– максимальний розмір блока даних, які передаватимуться; об'єм даних, що передаються без підтвердження про отримання;

– інтервали між передачами блоків даних та інтервал, протягом якого необхідно надіслати підтвердження;

– об'єм даних, який може прийняти кожна плата, не перепо-внюючись;

– швидкість передання даних.

Якщо новій (складнішій і швидшій) платі необхідно взаємодіяти із старою (повільною) платою, вони повинні знайти спільну

для обох швидкість передання. Схеми сучасних плат мережевого адаптера дозволяють їм пристосуватися до повільної швидкості старої плати.

Кожна плата оповіщає іншу про свої параметри, приймаючи «чужі» параметри і підстроюючись до них. Після того як всі деталі визначені, плата починає обмін даними.

Питання для самоперевірки

1. Охарактеризуйте найбільш важливі характеристики стандартних кабелів.
2. Назвіть види кабельних ліній.
3. Охарактеризуйте симетричні кабелі.
4. Охарактеризуйте коаксіальний кабель.
5. Назвіть й охарактеризуйте види ліній передач.
6. Назвіть основні споживчі параметри модемів.
7. Назвіть призначення плати мережного адаптера.

ТЕСТОВІ ЗАВДАННЯ

ТЕСТИ 1 РІВНЯ СКЛАДНОСТІ

1. З'ясуйте, для яких із указаних нижче цілей слугує команда *пересилки*?

- А. Для пересилки вмісту акумулятора в пам'ять.
- Б. Для пересилки вмісту одного регістра в інший.
- В. Для пересилки другого байта команди в регістр.
- Г. Для пересилки коду операції в дешифратор команд.
- Д. Для всіх випадків.

2. Визначте причину, за якою команди пересилки даних можна назвати командами копіювання даних?

А. Після пересилки дані залишаються в джерелі даних без змін.

- Б. Після пересилки даних вміст джерела даних знищується.
- В. Після пересилки даних вміст джерела даних змінюється.
- Г. У процесі пересилки дані змінюються.
- Д. У процесі пересилки дані не змінюються.

3. З'ясуйте, який пристрій не входить до структурної схеми мікропроцесорної системи?

- А. Порти введення/виведення.
- Б. Мікропроцесор.
- В. Контролери зовнішніх пристроїв.
- Г. Контролер прямого доступу до пам'яті.
- Д. Пристрій введення даних.

4. Програмна модель мікропроцесора – це:

- А. Опис програмно доступних регістрів мікропроцесора.
- Б. Опис всіх складових мікропроцесорної системи.
- В. Опис дій, які виконує арифметико-логічний пристрій мікропроцесора.
- Г. Опис прапорців регістра стану.
- Д. Опис команд мікропроцесора.

5. У якому випадку прапорець нульового результату в регістрі прапорців дорівнює одиниці?

- А. При виконанні арифметичних операцій.
- Б. При виконанні операцій зсуву.
- В. При отриманні нульового результату операції.
- Г. При отриманні в результаті операції одиниці.
- Д. При будь-якому результаті, який не дорівнює нулю.

6. У якому випадку прапорець негативного результату дорівнює одиниці?
- А. При виконанні дій над числами без знака.
 - Б. При виконанні дій над числами зі знаком.
 - В. При виконанні операції додавання.
 - Г. При отриманні одиниці в старшому розряді результату операції.
 - Д. При операціях із числами в додатковому коді.
7. Яке з наведених тверджень є правильним?
- А. Мови асемблера для різних обчислювальних систем не співпадають.
 - Б. Асемблери для різних обчислювальних систем у багатьох випадках схожі.
 - В. «Стандарт» асемблера для певної родини мікропроцесорів встановлюється авторами мікропроцесора та першим асемблером для нього.
 - Г. Для аналогічних команд в асемблерах можуть бути прийняті різні позначення.
 - Д. Усі твердження правильні.
8. Спільним принципом побудови пристроїв оброблення цифрової інформації є:
- А. Посторінкова організація простору пам'яті.
 - Б. У будь-якому пристрої оброблення цифрової інформації можна виділити дві частини: операційний і управляючий блоки.
 - В. Алгоритмом оброблення інформації є програма.
 - Г. Арифметичні та логічні перетворення даних здійснює арифметико-логічний пристрій.
 - Д. Програмне керування пристроєм оброблення цифрової інформації стало можливим з появою регістра стану.
9. У чому полягає процес виконання програми мікропроцесором?
- А. Процес виконання програми полягає в послідовному читанні кодів команд, що записані в пам'яті.
 - Б. Процес виконання програми складається з окремих машинних операцій, які здійснюють перетворення цифрової інформації під впливом машинної команди.
 - В. Процес виконання програми є перетворенням команд асемблера в машинні коди.
 - Г. Процес виконання програми є послідовністю сигналів пристрою управління.

Д. Процес виконання програми є реакцією мікропроцесорної системи на включення живлення.

10. Призначенням контролера прямого доступу до пам'яті є:

А. Читання інформації з дискети.

Б. Перетворення інформації.

В. Обмін даними між периферійними пристроями та пам'яттю без участі мікропроцесора.

Г. Аналіз сигналів переривань та організація реакції мікропроцесора на переривання.

Д. Усі перелічені функції.

11. Основною функцією контролера переривань є:

А. Приймання сигналів на переривання.

Б. Розподіл переривань на зовнішні і внутрішні.

В. Обмін даними між периферійними пристроями та пам'яттю без участі мікропроцесора.

Г. Аналіз сигналів переривань та організація реакції мікропроцесора на переривання.

Д. Усі перелічені функції.

12. У чому полягає процес функціонування мікропроцесорної системи?

А. В очікуванні сигналів на переривання.

Б. У виконанні команд програми та обміні даними.

В. У переданні інформації між периферійними пристроями та мікропроцесором.

Г. У перетворенні програми на асемблері в машинні коди.

Д. У перезаписуванні даних із постійного запам'ятовуючого пристрою в оперативну пам'ять.

13. Дайте визначення сегмента пам'яті?

А. Сегментом пам'яті є 8 біт.

Б. Сегментом пам'яті є декілька комірок пам'яті, необхідних для виконання поточної команди.

В. Сегментом пам'яті є область пам'яті, відведена під простір введення/виведення.

Г. Сегмент пам'яті є логічною одиницею пам'яті розміром 64 Кб.

Д. Сегментом пам'яті є перший мегабайт пам'яті.

14. Визначте мінімальну адресну одиницю пам'яті?

А. Один біт.

Б. Один байт.

- В. Один кілобайт.
- Г. Один мегабайт.
- Д. Один гігабайт.

15. Машинне представлення програми в пам'яті називають її об'єктним кодом. Наведіть його складові.

- А. Об'єктний код складається з коду операції та адресної частини.
- Б. Об'єктний код складається з нулів і одиниць.
- В. Об'єктний код складається з адрес регістрів загального призначення.
- Г. Об'єктний код складається з команд введення/виведення.
- Д. Об'єктний код складається з нулів, одиниць і команди NLT.

16. Визначте призначення процесора в ЕОМ.

- А. Процесор – це основний пристрій ЕОМ, який керує її роботою на основі аналізу програмних слів.
- Б. Процесор – це основний пристрій ЕОМ, який запам'ятовує результати виконаних операцій.
- В. Процесор – це основний пристрій ЕОМ, який виконує арифметичні і логічні операції.
- Г. Процесор – це основний пристрій ЕОМ, який пересилає слова даних із зовнішніх пристроїв до пам'яті.
- Д. Процесор – це основний пристрій ЕОМ, який аналізує зовнішні сигнали.

17. Які пристрої ЕОМ відносяться до основних (виконують основні функції)?

- А. Пристрій електроживлення.
- Б. Контролери зовнішніх пристроїв.
- В. Дисплей, клавіатура, миша.
- Г. Процесор, оперативний запам'ятовуючий пристрій, пристрої обміну даними з зовнішніми пристроями.
- Д. Контролер безпосереднього доступу до пам'яті.

18. Які типи сигналів використовуються у автоматизованих системах оброблення інформації?

- А. Двійкові.
- Б. Безперервні (аналогові) та дискретні (цифрові).
- В. Десяткові.
- Г. Шістнадцяткові.
- Д. Електричні сигнали напруги або струму.

19. З'ясуйте, у чому полягає основа роботи ЕОМ?

А. У перетворенні слів даних із десяткової системи числення в двійкову.

Б. У зберіганні слів даних у запам'ятовуючих пристроях.

В. У здатності виконання послідовності команд, які зберігаються в оперативному запам'ятовуючому пристрої.

Г. У здатності відображення інформації на дисплеї.

Д. У програмному забезпеченні.

20. З яких причин в ЕОМ використовується двійкова система числення?

А. Двійкова система числення мінімальна за кількістю цифр у розрядах.

Б. Двійкова система числення дозволяє використовувати зворотній код числа.

В. Двійкова система числення знайома всім із дитинства.

Г. Двійкова система числення легко запам'ятовується.

Д. Двійкова система числення дозволяє передавати інформацію в формі наявності (одиниця) або відсутності (нуль) сигналу.

21. З'ясуйте, із яких структурних елементів складається команда мікропроцесора?

А. З адреси регістра та акумулятора.

Б. З адрес пристроїв, між якими передаються слова даних.

В. Нулів і одиниць.

Г. Коду операції і адреси наступної команди.

Д. З коду операції і адресної частини.

22. Визначте, яку функцію виконує контролер прямого доступу до пам'яті?

А. Здійснює управління системою переривань.

Б. Організовує обмін даними між зовнішніми пристроями.

В. Управляє роботою мікропроцесора.

Г. Організовує обмін даними між зовнішніми пристроями і оперативною пам'яттю без участі мікропроцесора.

Д. Управляє роботою генератора тактової частоти.

23. Визначте призначення генератора тактової частоти?

А. Генератор тактової частоти виробляє тактові імпульси, які визначають швидкість роботи ЕОМ.

Б. Генератор тактової частоти приймає запити на переривання і в певний момент часу передає їх у мікропроцесор.

В. Генератор тактової частоти генерує сигнали, які рахують секунди в системному таймері.

Г. Генератор тактової частоти приймає сигнали з клавіатури.
Д. Генератор тактової частоти генерує сигнали при включенні живлення.

24. Який пристрій не входить до складу системної плати ЕОМ?

- А. Центральний процесор.
- Б. Оперативний запам'ятовуючий пристрій.
- В. Постійний запам'ятовуючий пристрій.
- Г. Контролер прямого доступу до пам'яті.
- Д. Жорсткий диск.

25. Який пристрій здійснює автономне живлення годинника реального часу в ЕОМ?

- А. Генератор тактової частоти.
- Б. Контролер шини.
- В. Мікропроцесор.
- Г. Літієва батарея.
- Д. Системний контролер.

26. Назвіть типи внутрішньої пам'яті ЕОМ?

- А. Оперативний запам'ятовуючий пристрій, постійний запам'ятовуючий пристрій, кеш-пам'ять.
- Б. Оперативний запам'ятовуючий пристрій, жорсткий диск, дискета.
- В. Жорсткий диск, дискета, флеш-пам'ять, CD-диск.
- Г. Виключно жорсткий диск.
- Д. Пам'ять програм.

27. Які пристрої відносять до зовнішньої пам'яті ЕОМ?

- А. Оперативний запам'ятовуючий пристрій, постійний запам'ятовуючий пристрій, кеш-пам'ять.
- Б. Оперативний запам'ятовуючий пристрій, жорсткий диск, дискета.
- В. Жорсткий диск, дискета, флеш-пам'ять, CD-диск.
- Г. Виключно жорсткий диск.
- Д. Пам'ять програм.

28. Для запису команди мікропроцесора використовують мнемонічну форму запису. Який із наведених прикладів є мнемонічною формою запису команд?

- А. 01100011.
- Б. 000111000111010.
- В. 303₈.

Г. JMP.

Д. 15H.

29. З якої причини використовують мнемонічний запис коду операції?

А. Мнемонічну форму використовують як скорочений запис назви операції, тому що її легко ототожнити з фактичним призначенням і легше запам'ятати, ніж команду, записану у двійковому коді.

Б. Мнемонічну форму використовують для скорочення запису коду операції до 3-х букв.

В. Мнемонічну форму використовують для спрощення роботи програми асемблера.

Г. Мнемонічну форму використовують для запису однакових програм для різних мікропроцесорів.

Д. Мнемонічну форму запису коду операції безпосередньо записують у пам'ять мікропроцесорного пристрою.

30. Визначте основну функцію програми асемблера.

А. Основною функцією програми асемблера є перетворення програми на мові асемблера в програму в машинних кодах.

Б. Основною функцією програми асемблера є перевірка логічних зв'язків між командами програми.

В. Основною функцією програми асемблера є перетворення програми в машинних кодах у програму з використання мнемонічного запису.

Г. Основною функцією програми асемблера є запис програми в пам'ять мікропроцесорної системи.

Д. Основною функцією програми асемблера є скорочення програмного коду.

31. З'ясуйте, що представляє собою адресна частина команди з безпосередньою адресацією?

А. Адресу області пам'яті в діапазоні від 0 до 255.

Б. 8-бітові дані.

В. Дані, які безпосередньо записує програміст у другий або другий і третій байт команди.

Г. Дані, які доступні будь-якій команді.

Д. Немає правильної відповіді.

32. На яку програму перетворюється програма, написана на будь-якій мові програмування?

А. На програму асемблера.

Б. На програму редактора.

В. На програму компілятора.

- Г. На програму в машинних кодах.
- Д. На програму інтерпретатора.

33. З'ясуйте, яка з мов програмування забезпечує перетворення однієї команди на цій мові на одну машинну команду?

- А. Pascal.
- Б. Асемблер.
- В. Мова кодових комбінацій.
- Г. Basic.
- Д. Усі наведені мови.

34. Припустимо, що у вашому розпорядженні є дві програми для вирішення задачі, записані мовою асемблера та мовою високого рівня. Яка з програм є універсальною (буде виконуватись на будь-якій ЕОМ)?

- А. Програма на мові асемблера.
- Б. Програма на мові високого рівня.
- В. І програма на мові асемблера, і програма на мові високого рівня.
- Г. Ніяка.
- Д. Програма мовою кодових комбінацій.

35. Послідовністю яких операцій можна описати процес множення?

- А. Процес множення можна описати послідовністю зсувів та додавань.
- Б. Процес множення можна описати порозрядним множенням і додаванням.
- В. Процес множення можна описати послідовністю зсувів і віднімань.
- Г. Процес множення можна описати додаванням.
- Д. Процес множення можна описати відніманням.

36. У якій формі частіше представляють негативні числа в ЕОМ?

- А. У зворотному коді.
- Б. У додатковому коді.
- В. У вигляді знака та модулю.
- Г. В абсолютному значенні.
- Д. У вигляді десяткового числа.

37. Визначте, у якому пункті наведені тільки негативні числа в додатковому коді?

- А. 11110000, 01010101, 00011100.

- Б. 10101010, 01010101, 00000000.
 В. 01001001, 11111111, 11100011.
 Г. 10000111, 11100010, 10111111.
 Д. Немає правильної відповіді.

38. До якого типу команд (логічних, арифметичних, управління, пересилки) мікропроцесора відноситься команда додавання ADD r?

- А. Логічних.
 Б. Арифметичних.
 В. Команд управління.
 Г. Пересилки.
 Д. Немає правильної відповіді.

39. До якого типу команд (логічних, арифметичних, управління, пересилки) мікропроцесора відноситься команда HLT?

- А. Логічних.
 Б. Арифметичних.
 В. Команд управління.
 Г. Пересилки.
 Д. Немає правильної відповіді.

40. До якого типу команд (логічних, арифметичних, управління, пересилки) мікропроцесора відноситься команда переходу JMP?

- А. Логічних.
 Б. Арифметичних.
 В. Команд управління.
 Г. Пересилки.
 Д. Немає правильної відповіді.

№	Відповідь	№	Відповідь	№	Відповідь	№	Відповідь
1	Б	11	Г	21	Д	31	В
2	А	12	Б	22	Г	32	Г
3	Д	13	Г	23	А	33	Б
4	А	14	Б	24	Д	34	Б
5	В	15	Б	25	Г	35	А
6	Г	16	А	26	А	36	Б
7	Д	17	Г	27	В	37	Г
8	Б	18	Б	28	Г	38	Б
9	Б	19	В	29	А	39	В
10	В	20	Д	30	А	40	А

ТЕСТИ 2 РІВНЯ СКЛАДНОСТІ

1. З'ясуйте призначення структурної схеми мікропроцесора?
 - А. Опис детальної конфігурації електронних компонентів з метою конструювання мікропроцесора.
 - Б. Опис логічного з'єднання мікропроцесора з пам'яттю та пристроями введення/виведення.
 - В. Опис логічних пристроїв, що входять до складу мікропроцесора, для програмування поставлених задач.
 - Г. Опис імен регістрів загального призначення.
 - Д. Опис регістра стану.

2. Для більшості арифметичних та логічних операцій є необхідним два операнда, один із яких знаходиться в пам'яті або в регістрі. У якому пристрої мікропроцесора знаходиться другий?
 - А. В акумуляторі.
 - Б. У лічильнику команд.
 - В. У регістрі адреси пам'яті.
 - Г. У регістрі команд.
 - Д. У регістрі загального призначення.

3. Регістром якого типу є лічильник команд?
 - А. Загального призначення.
 - Б. Спеціального призначення.
 - В. Буферним.
 - Г. Регістровою парою.
 - Д. Акумулятором.

4. Які дії виконує лічильник команд після зчитування коду команди з пам'яті?
 - А. Переходить до комірки пам'яті з початковою адресою.
 - Б. Збільшує своє значення на одиницю.
 - В. Чекає виконання поточної команди.
 - Г. Зменшує своє значення на одиницю.
 - Д. Не виконує ніякої дії.

5. Який пристрій мікропроцесора використовується для тимчасового збереження даних при переданні даних із зовнішнього порту в пам'ять?
 - А. Лічильник команд.
 - Б. Арифметико-логічний пристрій.
 - В. Акумулятор.
 - Г. Регістр загального призначення.
 - Д. Будь-який із наведених вище.

6. Мікропроцесорна система на базі 16-розрядного мікропроцесора має адресний простір 4 Мб. Скільки розрядів повинен мати регістр адреси пам'яті?

- А. 16.
- Б. 32.
- В. 64.
- Г. 22.
- Д. 8.

7. Яку з перелічених функцій арифметико-логічний пристрій не виконує?

- А. Додавання.
- Б. Віднімання.
- В. Включення живлення.
- Г. Зсув.
- Д. Логічне «і».

8. Назвіть критерій, за яким визначається розрядність регістра адреси пам'яті?

- А. Розрядність мікропроцесора.
- Б. Розрядність внутрішньої шини даних.
- В. Розрядність регістрів загального призначення.
- Г. Кількість необхідної пам'яті.
- Д. Сукупність перелічених критеріїв.

9. Яку роль можуть відігравати регістри загального призначення мікропроцесора?

- А. Регістрів стану.
- Б. Внутрішньої пам'яті мікропроцесора.
- В. Буферних регістрів.
- Г. Регістра команд.
- Д. Регістра адреси пам'яті.

10. У чому полягає основна функція арифметико-логічного пристрою?

- А. Виконувати операцію додавання.
- Б. Слугувати джерелом сигналів для акумулятора.
- В. Змінювати дані шляхом арифметичних і логічних операцій.
- Г. Слугувати сигналом початку наступної операції.
- Д. Усі перелічені функції.

11. Визначте розмір слова даних, яке можна зберігати в акумуляторі подвійної довжини 8-розрядного мікропроцесора?

- А. 1 байт.

- Б. 2 байти.
- В. 3 байти.
- Г. 4 байти.
- Д. Будь-який.

12. Адресний простір 16-розрядного мікропроцесора дорівнює 2^{16} , тобто 65536 байт. Підрахуйте кількість розрядів лічильника команд цього мікропроцесора.

- А. 8.
- Б. 16.
- В. 32.
- Г. 64.
- Д. 4.

13. На адресу якої команди вказує лічильник команд після вибірки з пам'яті чергової команди?

- А. Останню, що виконана.
- Б. Наступну.
- В. Чергову, що виконується.
- Г. Останню команду програми.
- Д. Першу команду програми.

14. Регістр адреси області пам'яті вказує на:

- А. Вміст області пам'яті.
- Б. Адресу наступної команди.
- В. Регістр внутрішньої пам'яті.
- Г. Адресу області пам'яті.
- Д. Акумулятор.

15. Регістр адреси області пам'яті сполучається з внутрішньою шиною даних мікропроцесора. Який із наступних об'єктів слугує джерелом для цього регістра?

- А. Лічильник команд.
- Б. Дешифратор команд.
- В. Акумулятор.
- Г. Регістр стану.
- Д. Усі перелічені об'єкти.

16. Процес виконання команди має дві стадії: 1 – вибірка; 2 – виконання. Адреса якої команди зберігається в регістрі команд на стадії виконання поточної команди?

- А. Поточної команди.
- Б. Наступної команди.
- В. Регістр порожній.

- Г. Першої команди програми.
- Д. Останньої команди програми.

17. Яке основне призначення буферних регістрів?

- А. Поєднувати арифметико-логічний пристрій із пам'яті.
- Б. Виконувати операцію додавання.
- В. Слугувати для акумулятора запам'ятовуючим пристроєм.
- Г. Розподіл у часі сигналів на вході і виході арифметико-логічного пристрою.
- Д. Виконувати операцію зсуву.

18. Робота мікропроцесорної системи синхронізується сигналами тактової частоти. З'ясуйте їх призначення.

- А. Інформувати мікропроцесор про поточний час.
- Б. Керувати роботою запам'ятовуючих пристроїв.
- В. Перевіряти наявність даних у регістрах.
- Г. Визначати послідовність сигналів пристрою управління в часі.
- Д. Визначати моменти часу для виконання команд.

19. Команда мікропроцесора складається з двох частин: перша – код операції, друга – адресна частина. Про що інформує мікропроцесор код операції?

- А. Що робити;
- Б. Де виконувати задані дії.
- В. Де зберігаються дані, необхідні для заданої операції.
- Г. Як змінюється регістр стану після виконання команди.
- Д. Де зберігається результат операції.

20. Визначте, чому дорівнює довжина команди 8-розрядного мікропроцесора?

- А. 1 байт.
- Б. 2 байти.
- В. 1, 2 або 3 байти.
- Г. Довжина команди не залежить від розрядності мікропроцесора.
- Д. Може бути будь-якою.

21. Команда мікропроцесора складається з двох частин: перша – код операції, друга – адресна частина. Про що інформує мікропроцесор адресна частина команди?

- А. Що робити.
- Б. Де виконувати задані дії.
- В. Де зберігаються дані, необхідні для заданої операції.

- Г. Як змінюється реєстр стану після виконання команди.
- Д. Де зберігається результат операції.

22. З'ясуйте, які дані зберігаються адресною частиною команди з безпосередньою адресацією?

- А. Адреса області пам'яті в діапазоні від 010 до 25510.
- Б. Безпосередньо дані, які беруть участь в операції.
- В. Адреси реєстрів загального призначення.
- Г. Адреса пристрою введення/виведення?
- Д. Число п.

23. Непряма адресація в командах мікропроцесора ще має назву реєстрова, тому що

- А. Адресна частина команди зберігає безпосередньо дані, необхідні для оброблення.
- Б. Визначена реєстрова пара мікропроцесора зберігає адресу даних.
- В. Адреса даних зберігається в акумуляторі.
- Г. Адресна частина команди зберігає адреси портів введення/виведення.
- Д. Адресна частина команди відсутня.

24. Визначте, який вид інформації передається по лініях внутрішньої шини даних мікропроцесора.

- А. Слова даних.
- Б. Адреси комірок пам'яті.
- В. Сигнали від пристрою управління.
- Г. Вміст комірок зовнішньої пам'яті.
- Д. Дані про стан мікропроцесора.

25. Визначте, який вид інформації передається по лініях шини управління.

- А. Слова даних.
- Б. Адреси комірок пам'яті.
- В. Сигнали від пристрою управління.
- Г. Вміст комірок зовнішньої пам'яті.
- Д. Дані про стан мікропроцесора.

26. Визначте, який вид інформації передається по лініях шини адреси мікропроцесора.

- А. Слова даних.
- Б. Адреси комірок пам'яті або пристроїв введення/виведення.
- В. Сигнали від пристрою управління.
- Г. Вміст комірок зовнішньої пам'яті.

Д. Дані про стан мікропроцесора.

27. Мікропроцесорна система на базі 8-розрядного процесора має чотири порти введення/виведення, кожний із яких має по вісім ліній даних. Яку назву мають такі порти?

- А. Паралельні порти введення/виведення.
- Б. Послідовні порти введення/виведення.
- В. Порти виведення адрес.
- Г. Універсальні порти.
- Д. Порти принтера.

28. Які з перелічених сигналів передаються по шині управління?

- А. «Скидання».
- Б. «Читання пам'яті».
- В. «Запис у пам'ять».
- Г. «HLT».
- Д. Усі перелічені сигнали.

29. Визначте, які з перелічених характеристик правильні по відношенню до адресної шини мікропроцесора.

- А. Кількість ліній шини адреси дорівнює восьми.
- Б. Кількість ліній шини адреси дорівнює шістнадцяти.
- В. Шина адреси передає дані у двох напрямках.
- Г. Шина адреси є однонаправленою.
- Д. Шина адреси є універсальною.

30. Визначте, які з перелічених характеристик правильні по відношенню до шини даних мікропроцесора?

- А. Кількість ліній шини даних дорівнює восьми.
- Б. Кількість ліній шини даних дорівнює шістнадцяти.
- В. Шина даних передає дані у двох напрямках.
- Г. Шина даних є однонаправленою.
- Д. Шина даних є універсальною.

№	Відповідь	№	Відповідь	№	Відповідь
1	В	11	Б	21	Б
2	А	12	Б	22	Б
3	Б	13	Б	23	Б
4	Б	14	Г	24	А
5	В	15	А	25	В
6	Б	16	Б	26	Б
7	В	17	Г	27	А
8	Д	18	Г	28	Д
9	Б	19	А	29	Г
10	В	20	В	30	В

ТЕСТИ 3 РІВНЯ СКЛАДНОСТІ

1. Визначте причину зручності використання додаткового коду числа в ЕОМ.

А. Одні й ті ж логічні схеми можна використовувати для додавання і віднімання двійкових чисел без знака і додаткових кодів негативних чисел.

Б. Запис чисел у додатковому коді не відрізняється від запису чисел у двійковому коді.

В. Операція отримання додаткового коду легко реалізується апаратно.

Г. Додатковий код числа легко перетворюється на двійковий.

Д. Немає правильної відповіді.

2. Визначте синонім терміна «регістр стану»?

А. Регістр адреси пам'яті.

Б. Регістр прапорців.

В. Акумулятор.

Г. Регістрова пара.

Д. Регістр команд.

3. З'ясуйте, що відбувається при додаванні двох двійкових чисел, якщо число, що є їх сумою, записується двома цифрами?

А. Позика.

Б. Перенос.

В. Отримання зворотного коду.

Г. Додавання.

Д. Віднімання.

4. До якого типу команд (логічних, арифметичних, управління, пересилки) мікропроцесора відноситься команда зсуву?

А. Логічних.

Б. Арифметичних.

В. Команд управління.

Г. Пересилки.

Д. Немає правильної відповіді.

5. Визначте, яку дію треба виконати при програмуванні алгоритмів з двійково-десятковими числами (числами в коді 8421)?

А. Отримання додаткового коду.

Б. Отримання зворотного коду.

В. Двійково-десятькова корекцію.

Г. Попередньо перетворювати числа у двійкову систему числення.

Д. Попередньо перетворювати числа в десяткову систему числення.

6. Яка з перелічених дій виконується при виконанні команди «прирошення позитивне»?

А. Додавання одиниці до вмісту регістра.

Б. Віднімання одиниці із вмісту регістра.

В. Додавання біта переносу в молодший розряд регістра.

Г. Додавання біта нульового результату в молодший розряд регістра.

Д. Запис у регістр нуля.

7. Яка з перелічених дій виконується при виконанні команди «прирошення негативне»?

А. Додавання одиниці до вмісту регістра.

Б. Віднімання одиниці із вмісту регістра.

В. Додавання біта переносу в молодший розряд регістра.

Г. Додавання біта нульового результату в молодший розряд регістра.

Д. Запис у регістр нуля.

8. Яка з перелічених дій виконується при виконанні команди введення даних IN?

А. Додавання одиниці до вмісту регістра команд.

Б. Віднімання одиниці із вмісту регістра команд.

В. Запис у регістр В слова даних із вхідного порту.

Г. Запис у регістр С слова даних із вхідного порту.

Д. Запис в акумулятор слова даних із вхідного порту.

9. Яка з перелічених дій виконується при виконанні команди виведення даних OUT?

А. Додавання одиниці до вмісту регістра команд.

Б. Віднімання одиниці із вмісту регістра команд.

В. Передання слова даних з регістра В до вихідного порту.

Г. Передання слова даних з регістра С до вихідного порту.

Д. Передання слова даних з акумулятора до вихідного порту.

10. Визначте, як працює мікропроцесор з портом введення/виведення при організації звернення до зовнішніх портів за аналогією зі зверненням до пам'яті.

А. Як з регістром загального призначення.

Б. Як з акумулятором.

В. Як з адресою в області пам'яті.

Г. Як із клавіатурою.

Д. Немає правильної відповіді.

11. Яку дію не виконує мікропроцесор при надходженні сигналу на переривання?
- А. Закінчує виконання поточної команди.
 - Б. Поточне значення лічильника команд завантажує в стек.
 - В. У лічильник команд завантажує адресу програми оброблення переривань.
 - Г. Завантажує в стек значення акумулятора.
 - Д. Немає правильної відповіді.
12. Яка система числення найчастіше використовується для адресації пам'яті?
- А. Двійкова.
 - Б. Вісімкова.
 - В. Десяткова.
 - Г. Шістнадцяткова.
 - Д. Немає правильної відповіді.
13. Яка система числення найчастіше використовується для введення даних із клавіатурних пристроїв?
- А. Двійкова.
 - Б. Вісімкова.
 - В. Десяткова.
 - Г. Шістнадцяткова.
 - Д. Немає правильної відповіді.
14. Яка система числення використовується в регістрі пропорцій?
- А. Двійкова.
 - Б. Вісімкова.
 - В. Десяткова.
 - Г. Шістнадцяткова.
 - Д. Немає правильної відповіді.
15. Яка система числення використовується для адресації портів введення/виведення?
- А. Двійкова.
 - Б. Вісімкова.
 - В. Десяткова.
 - Г. Шістнадцяткова.
 - Д. Немає правильної відповіді.
16. Яка система числення використовується в акумуляторі?
- А. Двійкова.
 - Б. Вісімкова.

- В. Десяткова.
- Г. Шістнадцяткова.
- Д. Немає правильної відповіді.

17. Яка система числення використовується в арифметико-логічному пристрої?

- А. Двійкова.
- Б. Вісімкова.
- В. Десяткова.
- Г. Шістнадцяткова.
- Д. Немає правильної відповіді.

18. Яку першу дію виконує мікропроцесор при надходженні сигналу на переривання?

- А. Закінчує виконання поточної команди.
- Б. Поточне значення лічильника команд завантажує в стек;
- В. У лічильник команд завантажує адресу програми оброблення переривань.
- Г. Завантажує в стек значення акумулятора.
- Д. Виконує першу команду програми переривання.

19. Яку дію виконує мікропроцесор при закінченні виконання програми переривання?

- А. Закінчує виконання поточної команди.
- Б. Поточне значення лічильника команд завантажує в стек.
- В. У лічильник команд завантажує адресу програми оброблення переривань.
- Г. У лічильник команд завантажує із стека адресу команди основної програми, яка повинна виконуватись наступною.
- Д. Виконує першу команду основної програми.

20. Де зберігаються адреси перших команд програм переривання?

- А. У таблиці векторів переривань.
- Б. У комірках пам'яті з адресами 01Н до 21Н.
- В. У регістрі прапорців.
- Г. У стеку.
- Д. У лічильнику команд.

21. З'ясуйте призначення команди «десяткова корекція»?

- А. Перетворення числа з десятикової системи числення у двійкову.
- Б. Перетворення числа з десятикової системи числення у двійково-десятковий код.

В. Введення поправки в результат додавання двох двійково-десяткових чисел.

Г. Перетворення числа у двійково-десятковому коді в десяткову систему числення.

Д. Введення поправки в результат віднімання двох двійково-десяткових чисел.

22. З'ясуйте, який пристрій зберігає еталон при виконанні команди «порівняння»?

А. Акумулятор.

Б. Лічильник команд.

В. Адресна частина команди.

Г. Пам'ять.

Д. Регістр загального призначення.

23. Команда мікропроцесора – це слово, що однозначно сприймається внутрішніми схемами, які керують роботою ЕОМ. Як відображається це слово в пам'яті?

А. У двійковій системі числення.

Б. У шістнадцятковій системі числення.

В. У десятковій системі числення.

Г. На мові асемблера.

Д. У вісімковій системі числення.

24. Яка послідовність запису даних в адресній частині команди?

А. Старший байт адреси або даних, молодший байт адреси або даних.

Б. Молодший байт адреси або даних, старший байт адреси або даних.

В. Старший байт коду операції, молодший байт коду операції.

Г. Молодший байт коду операції, старший байт коду операції.

Д. Немає правильної відповіді.

25. У якій системі числення представлено число 2AFD?

А. У двійковій системі числення.

Б. У шістнадцятковій системі числення.

В. У десятковій системі числення.

Г. На мові асемблера.

Д. У вісімковій системі числення.

26. Визначте зміст поняття «скидання» акумулятора?

А. Встановлення всіх розрядів акумулятора в нуль.

Б. Встановлення всіх розрядів акумулятора в одиницю.

- В. Завершення використання акумулятора.
- Г. Виведення вмісту акумулятора на зовнішній пристрій.
- Д. Введення в акумулятор слова даних із зовнішнього пристрою.

27. Яку кількість розрядів повинен мати буферний регістр 16-розрядного мікропроцесора?

- А. 32.
- Б. 16.
- В. 8.
- Г. 4.
- Д. Будь-яку кількість розрядів.

28. Якою шиною керує вихід регістра адреси пам'яті?

- А. Шиною управління.
- Б. Шиною даних.
- В. Шиною адреси.
- Г. Шиною сигналів.
- Д. Шиною заземлення.

29. З'ясуйте, що поєднує всі функціональні вузли мікропроцесора?

- А. Схеми управління.
- Б. Внутрішня шина даних.
- В. Акумулятор.
- Г. Лічильник команд.
- Д. Шина адреси.

30. З'ясуйте, що є входом і виходом регістра команд?

- А. Схеми управління.
- Б. Внутрішня шина даних і дешифратор команд.
- В. Акумулятор і регістр стану.
- Г. Лічильник команд.
- Д. Шина адреси.

№	Відповідь	№	Відповідь	№	Відповідь
1	А	11	Г	21	В
2	Б	12	Г	22	А
3	Б	13	В	23	А
4	Б	14	А	24	А
5	В	15	Г	25	Б
6	А	16	А	26	А
7	Б	17	А	27	Б
8	Д	18	А	28	В
9	Д	19	Г	29	Б
10	В	20	А	30	Б

Критерії оцінювання тестових завдань (за рівнями складності)

Оцінка	Кількість правильних відповідей		
	1 рівень складності	2 рівень складності	3 рівень складності
«Відмінно»	35–40	26–30	27–30
«Добре»	24–34	19–25	22–26
«Задовільно»	12–23	15–18	18–21
«Незадовільно»	0–11	0–14	0–17

* Час виконання завдання становить 2 академічні години

Предметний покажчик

адреса

– мережева 256

– слів 21

– базова 74

акумулятор 32, 35, 92

арифметико-логічний пристрій 27

архітектура 214

– ЕОМ 10

– клієнт-сервер 215 220

– комп'ютерних мереж 212, 216

– однорангова 215, 218

– процесора 32

– системи 10

– термінал-головний комп'ютер 215, 217

аудіоадаптер 194, 199

відеоадаптер 175

віртуальна пам'ять 28

великі інтегральні схеми 11

волокно

– оптичне 247

– багатомодове 248

дані

– цифрові 252

декомпресія

– відеосигналів 198

– звукових сигналів 201

диск

– лазерний 202

– оптичний 196

електричний шум 235

електронно-променева трубка 166

загасання 240

– коефіцієнт 241

звукова карта 199

зовнішні запам'ятовуючі пристрої 28

імпеданс 234

інтерфейс 28, 217

інформаційна система 11, 213

інформація 11

кабель 233, 236

– волоконно-оптичні лінії зв'язку 209

– екранована вита пара 243

– коаксіальний 244

– симетричний 237

кадр

– оцифрований 195

– частота 169

канал

– зв'язку 52, 59, 208

– логічний 214

– фізичний 214

кеш-пам'ять 159, 161

клавіатура 189

– герконова 190

– мембранна 190

– механічна 190

– напівмеханічна 190

клавіша

– навігаційна 192

– функціональна 192

команда 21, 63

компресія 196

– звукових сигналів 198, 199

контролер 28, 57

– пам'яті 163

лічильник команд 34, 78, 94

материнська плата, 158

машинне слово 15

мікроконтролери 51

мікропроцесор 8, 11

– універсальний 52

– сигнальний 52

мікропроцесорна система 11

мікропрограмне управління 23

мережа 212

– глобальна 212

– інформаційна 212

– комунікаційна 212

– локальна 212

- національна 212
- регіональна 212
- розподілена 212
- територіальна 212
- транснаціональна 212
- мікросхеми пам'яті 161
- статичні 161
- динамічні 161
- місткість 115, 193, 242
- метод
- доступу 214
- стискання даних 209
- розгортання даних 194
- фрактальний 197
- модем 252
- монітор 165
- МОП-технології 89
- мультимедіа 194

надвеликі інтегральні схеми 35

накопичувач на лазерному диску 158

носії 193

об'єднувальна плата 158, 159

- пасивна 158
- активна 159

обчислювальна система 12

ОЗП 26, 27, 56

- статичні 161

операнди 23, 35

операційна система 131

опір 234

- активний 235
- хвильовий 234

пам'ять

- динамічна 164
- енергетична 164
- зовнішня 20, 44
- оперативна 27, 160

перетворювач

- аналого-цифровий 198
- цифро-аналоговий

плата

- мережевого адаптера 254
- показчик стека 76, 99
- принтер 158, 182
- пристрій
- управління 26, 33, 64
- програма 55, 63, 69, 113
- мережева 222
- програмовані логічні інтегральні схеми 27
- протокол 85, 200, 214, 238
- процесор 20, 27
- центральний
- процесорний модуль 159
- п'єзоефект 186

- регістр
- адреси пам'яті 32
- загального призначення 32
- команд 34
- сегментний 148
- стану 32, 92
- режим роботи процесора 111
- роздільна здатність 208
- розрядність шини 101

- сервер 220
- сервіс 220
- системна плата 158
- сигнал 12
- синтез 82, 199
- синтезатор 199
- синхронізація 64, 195
- сканер 9, 158
- слоти 158, 254
- сопла 185
- стандарт
- EIA/TIA-568A 233
- EIA/TIA-568 233
- TSB-36 233
- TSB-40A 233
- ISO/IEC 11801 233
- EN50173 233
- стан процесора 39
- стрімер 43
- структура

- адресної пам'яті 45
- асоціативної пам'яті 47
- команд 70
- мікроконтролерів 83
- мікропроцесорного пристрою 55
- системи 10
- системний блок 158
- схема
 - інтерфейсна 105
 - синхронізації 159

таблиця дескрипторів 126
 такт 31, 86
 топографія 191
 топологія 223

- мережі 223
- шинна 224
- зірка 226
- кільце, 229
- шинно-зіркоподібна 231
- зіркоподібна-кільцева 231

 транзистори 89
 трафік 214
 ТСО 178

фізична адреса 134
 флеш-пам'ять 193, 254

час доступу 42, 110, 163
 частота

- тактова 31

 числа

- з рухомою крапкою 17

швидкість

- поширення 241

 шина

- комп'ютера 256

Ядро

- процесорне 85
- інтегральної схеми 161

adtlition 182

Advanced Interactive Video 207
Advanced Signal Processor 201
AT 159
ATX 159
aperture grill 168
Apple 160
Attenuation 234

Baby-AT 159
Backplane 158
Bandwidth 169
BEDO 164
Bubble-jet 185

capture board, 195
CAS, 163
CD-DA, 203
CD-ROM, 158
Centering 171
client-server architecture 220
CMOS Setup 164
Column 161
Column Access Strobe 163
Color Graphics Adapter 176
colour laser 184
Compaq 160
CMYK 183
CPU 27, 164
CromaClear 168
CRT 166
Constant Angular Velocity 202
Constant Linear Velocity 202
continuous drop 185
continuous jet 185

DDR SDRAM 165
Density 161
Digital Signal Processor 201
Differential Pulse Code Modulation 197
DIMM 164
Discrete Cosine Transform 197
dot pitch 167
DRAM 164
drop-on-demand 185

Dye sublimation 184

EDO 164

Encoder 195

Enhanced Graphics Adapter 176

Fiber Distributed Data Interface 230

Field Emission Display 174

Flat Panel Displays 181

FPM 164

frame grabbers 195

Genlock 195

High end 185

Hardware 157

IBM 160

Intel 25, 32, 109

Iris Graphics 185

JAZ-накопичувач 193

Laptop i Notebook 181

Liquid Crystal Display 169, 175

liquid ink-jet 184

LPX 159

Luminoform 166

main board 158

mothercard 158

motherboard 158

MPRII 178

multiport repeater 227

Musical Instruments Digital Interface 200

Near End Cross Talk (NEXT) 234

NLX 159

Notebook 181

PC 94

Pentium 109

Pentium Pro 130

phase-change ink-jet 184

Plasma Addressed Liquid Crystal 173
piezo 185
pixel – picture element 182

Random Access Memory (RAM) 162
RAS 163
RDRAM 165
Row 161, 163
Row Access Strobe 163

SDRAM 164
SIMM 164
SIP 164
slot mask 167
slot pitch 168
Sound Blaster 198
System Network Architecture (SNA) 218

terminal – host computer architecture 217
The Swedish Board for Technical Accreditation 181
The Swedish Confederation of Professional Employees 179
thermal ink-jet 185
thermal wax transfer 184, 187

Video Graphics Array 176

WORM 203
WR 163
Write 163

ZIP-накопичувач 193

Література

1. Гук М. Процессоры Pentium II, Pentium Pro и просто Pentium / М. Гук. – СПб. : Питер, 1999. – 288 с.
2. Гук М. Аппаратные средства локальных сетей : энциклопедия / М. Гук. – СПб. : Питер, 2000. – 572 с.
3. Гук М. Дисковая подсистема ПК / М. Гук. – СПб. : Питер, 2001. – 275 с.
4. Рудометов Е. Материнские платы и чипсеты / Е. Рудометов, В. Рудометов. – 3-е изд. – СПб. : Питер, 2002. – 352 с.
5. Гук М. Аппаратные интерфейсы ПК : энциклопедия / М. Гук. – СПб. : Питер, 2007. – 448 с.
6. Гук М. Аппаратные интерфейсы ПК : энциклопедия / М. Гук. – СПб. : Питер, 2002. – 528 с.
7. Вильховченко С. Современный компьютер: устройство, выбор, модернизация / С. Вильховченко. – СПб. : Питер, 2000. – 512 с.
8. Гук М. Аппаратные средства IBM PC : энциклопедия / М. Гук. – 3-е изд. – СПб. : Питер, 2006. – 1072 с.
9. Левин М. Компьютерные сети: устройство, подключение и использование / М. Левин. – М. : Оверлей, 2001. – 416 с.
10. Новиков Ю. Персональные компьютеры: аппаратура, системы, Интернет : учеб. курс / Ю. Новиков, А. Черепанов. – СПб. : Питер, 2001. – 464 с.
11. Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы / В. Г. Олифер, Н. А. Олифер. – 3-е изд. – СПб. : Питер, 2006. – 958 с.
12. Ричков В. Компьютер для студента : самоучитель / В. Ричков, В. Дьяконов, Ю. Новиков. – СПб. : Питер, 2000. – 592 с.
13. Калиш Г. Г. Основы вычислительной техники / Г. Г. Калиш. – М. : Высш. шк. 2000. – 269 с.
14. Скляр А. Я. Архитектура, принципы функционирования и управление ресурсами IBM PC / А. Я. Скляр. – Х. : Бизнес Информ, 2001. – 420 с.
15. Калабеков Б. А. Цифровые устройства и микропроцессорные системы / Б. А. Калабеков. – М. : Горячая линия ; Телеком, 2000. – 336 с.
16. Гилмор Ч. Введение в микропроцессорную технику / Ч. Гилмор. – М. : Мир, 1984. – 340 с.
17. Гуров В. В. Основы теории и организации ЭВМ : учеб. пособие / В. В. Гуров, В. О. Чуканов. – М. : ИНТУИТ.РУ, 2006. – 272 с.
18. Мікропроцесорна техніка : підручник / Ю. І. Якименко [та ін.]. – К. : Політехніка ; Кондор, 2004. – 352 с.
19. Схемотехніка електронних систем : у 3 кн. Кн. 3. Мікропроцесори та мікроконтролери : підручник / В. І. Бойко, А. М. Гуржій, В. Я. Жуйков [та ін.]. – 2-е вид. – К. : Вища шк., 2004. – 399 с.
20. Архітектура інформаційних мереж [Електронний ресурс]. – Режим доступу: <http://prvo.dynalias.net/book/archnet>.
21. Комп'ютерні мережі [Електронний ресурс]. – Режим доступу: <http://comp-net.at.ua/index>.

ПЕРЕДМОВА.....	3
Перелік умовних скорочень.....	6
Розділ 1 Основи ЕОМ.....	8
1.1. Огляд предметної області.....	8
1.2. Основні поняття теорії ЕОМ.....	9
1.3. ЕОМ як пристрій для оброблення цифрової інформації.....	11
1.4. Форми представлення інформації в ЕОМ.....	13
1.4.1. Двійкова система числення.....	14
1.4.2. Представлення чисел з фіксованою і рухомою крапкою	16
Питання для самоперевірки.....	17
Розділ 2 Принципи побудови електронно-обчислювальних машин.....	18
2.1. Принцип дії ЕОМ	18
2.2. Характеристики ЕОМ	22
2.3. Класична схема ЕОМ.....	23
2.4. Магістрально-модульний принцип організації ЕОМ	24
2.5. Архітектурна організація процесора ЕОМ	27
2.6. Принципи організації системи переривань програм	33
2.7. Поняття про стан процесора (програми). Вектор (слово) стану	35
2.7. Загальні відомості і класифікація пристроїв пам'яті.....	37
Питання для самоперевірки.....	44
Розділ 3 Мікропроцесорна система та її функціонування.....	46
3.1. Класифікація мікропроцесорів.....	46
3.2. Типова структура мікропроцесорного пристрою	49
3.3. Способи передавання цифрової інформації в мікропроцесорній системі	51
3.4. Функціонування мікропроцесорної системи.....	53
3.5. Організація просторів пам'яті та введення/виведення.....	54
3.6. Командний цикл	57
3.7. Принципи організації та класифікація арифметико-логічних пристроїв	59
3.8. Проектування команд	60
3.9. Структура та класифікація команд.....	62
3.10. Адресація в командах мікропроцесора	64
3.11. Адресація стека.....	67
3.12. Команди, процедури і мікропрограми передавання управління	69
Питання для самоперевірки.....	70
Розділ 4 Мікроконтролери	71
4.1. Цифрові пристрої на основі мікроконтролерів	71
4.2. Класифікація і структура мікроконтролерів.....	73
4.3. Процесорне ядро мікроконтролера та його структура	75
4.4. Приклади мікроконтролерів.....	78
4.4.1. Мікроконтролер i8051	78
4.4.2. Структурна організація AVR-мікроконтролера	88

4.5. Програмування мікроконтролерів.....	91
4.6. Периферійні інтерфейси і можливості зв'язку	92
Питання для самоперевірки	94
Розділ 5 Мікропроцесор.....	95
5.1. Мікропроцесор типу Intel.....	95
5.2. Програмна модель процесорів типу IA-32	98
5.3. Організація пам'яті	113
5.4. Функціональна класифікація машинних команд.....	121
5.5. Система команд процесора	126
5.6. Структура машинної команди мікропроцесорів Intel 80x86	129
5.7. Способи завдання операндів команди (способи адресації).....	133
Питання для самоперевірки	136
Розділ 6 АРХІТЕКТУРА ПЕРСОНАЛЬНОГО КОМП'ЮТЕРА	137
6.1. Апаратні засоби персонального комп'ютера	137
6.2. Системний блок.....	138
6.3. Монітор	144
6.4. Технології друкуючих пристроїв	159
6.5. Пристрої введення даних	166
6.6. Пристрої зберігання даних.....	169
6.7. Засоби мультимедіа	170
6.7.1. Відео і аудіо засоби	171
6.7.2. Носії інформації.....	177
6.8. Периферійні пристрої для систем спостереження	180
Питання для самоперевірки	185
Розділ 7 Архітектура комп'ютерних мереж	186
7.1. Архітектура «термінал – головний комп'ютер».....	190
7.2. Однорангова архітектура	191
7.3. Архітектура «клієнт – сервер».....	192
7.4. Базові топології мереж	195
7.5. Принцип функціонування шинної топології.....	196
7.6. Зіркоподібна топологія: принцип функціонування.....	198
7.7. Мережі з кільцевою топологією.....	200
7.8. Змішані топології: шинно-зіркоподібна і зіркоподібно-кільцева.....	201
Питання для самоперевірки	202
Розділ 8 Фізичне середовище передання	203
8.1. Мережний кабель	203
8.2. Симетричні кабелі.....	206
8.3. Коаксіальний кабель	212
8.4. Волоконно-оптичні лінії зв'язку	216
8.5. Пристрої обміну даними	220
8.6. Плати мережевого адаптера.....	222
Питання для самоперевірки	225
Тестові завдання	226
Тести 1 рівня складності	226
Тести 2 рівня складності	235

Тести 3 рівня складності.....	241
Предметний показник	248
Література.....	256

Сезонова І. К.

С 28 Архітектура ЕОМ і мікропроцесорні системи : навч. посіб. / І. К. Сезонова, Т. П. Колісник, Ю. Е. Хорошайло ; МВС України, Харк. нац. ун-т внутр. справ, 2011. – 260 с.
ISBN 978-966-8689-16-1

Навчальний посібник призначений для студентів та курсантів, які навчаються за напрямом «комп'ютерні науки». У посібнику розглянуті теоретичні основи мікропроцесорної техніки, склад та принцип роботи процесорів, мікроконтролерів, пам'яті, шин, запам'ятовуючих пристроїв, моніторів, принтерів і т. ін. Розглянуто питання, що стосуються нових периферійних пристроїв, поняття про мережі, їх призначення і організацію. Після кожного розділу є список контрольних питань, які допоможуть перевірити вивчений матеріал. Для самоперевірки надані тестові завдання трьох рівнів складності з таблицями відповідей.

УДК 004.2+004.27
ББК 32.973.2-02+32.973.26-04

Навчальне видання

СЕЗОНОВА Ірина Костянтинівна
КОЛІСНИК Тетяна Петрівна
ХОРОШАЙЛО Юрій Євгенович

АРХІТЕКТУРА ЕОМ І МІКРОПРОЦЕСОРНІ СИСТЕМИ

Навчальний посібник

Редактор *Н.С.Леонова*
Комп'ютерне верстання *С.В.Гончарук*

Формат 60х84/16. Ум.-друк.арк. 15,17. Обл.-вид.арк. 12, 35.
Тираж 300 пр. Зам. № 1212-3.

Віддруковано у ТОВ «Оберіг»
61140, м. Харків, просп. Гагаріна, 62, к.97.
Свідоцтво про держреєстрацію ДК № 3045 від 07.12.2007