

## ТЕМА 7. Мікропроцесори

### 7.1 Організація шин

#### 7.2 Принципи побудови мікропроцесорних систем

#### 7.3 Архітектура мікропроцесорів

### 7.1. Організація шин

*Шина* - це інформаційний канал, який об'єднує всі функціональні блоки МПС і забезпечує обмін даними у вигляді двійкових чисел. Конструктивно шина являє собою  $n$  провідників та один спільний провідник (земля). Дані про шину передаються у вигляді слів, що є групою бітів.

У паралельній шині  $n$  бітів передаються по окремих лініях одночасно, у послідовній шині - по єдиній лінії послідовно у часі. Паралельні шини виконують у вигляді плоского кабелю, а послідовні - у вигляді коаксіального або волоконно-оптичного кабелю. Коаксіальний кабель використовують для передачі даних на відстань до 100 метрів, узгоджуючи передавальні та приймальні каскади із хвильовим опором лінії. Волоконно-оптичний кабель використовують для передачі на більші відстані.

Усі основні блоки МПС з'єднують з єдиною паралельною шиною, яка називається *системною шиною*. Системна шина містить три шини: адреси, даних і керування.

*Шина адреси* є однонапрямленою. Вона призначена для передавання адреси комірки пам'яті або пристрою введення-виведення (ПВВ). Напрямок передавання по шині адреси - від МП до зовнішніх пристроїв. Варіанти умовних позначень однонапрямленої паралельної шини показано на рис. 7.1, на якому стрілка вказує напрям передавання.

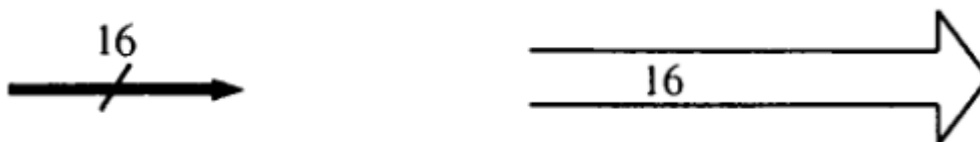


Рис. 7.1. Варіанти умовних позначень однонапрямленої паралельної 16-розрядної шини

Число 16 на рис. 7.1 позначає розрядність шини. Зазначимо, що допускається позначення шин і без наведення розрядності.

*Шина даних* є двонапрямленою. Вона призначена для передавання даних між блоками МПС. Інформація по одних і тих самих лініях ОВ

може передаватися у двох напрямках – як до МП, так і від нього. Варіанти умовних позначень двонапрявленої шини показано на рис. 7.2.

*Шина керування* призначена для передавання керувальних сигналів. Хоча напрям керувальних сигналів може бути різним, однак шина керування не є двонапрявленою, оскільки для сигналів різного напрямку використовуються окремі лінії. Позначається ця шина так само, як і однонапрявлена (див. рис. 7.1).

Як приклад на рис. 7.3 показано структурну схему передавання інформації між  $m$  регістрами по внутрішній  $n$ -розрядній шині даних з урахуванням прийнятих позначень, а на рис. 7.4 – розширену структурну схему.



Рис. 7.2. Варіанти умовних позначень двонапрявленої паралельної 8-розрядної шини

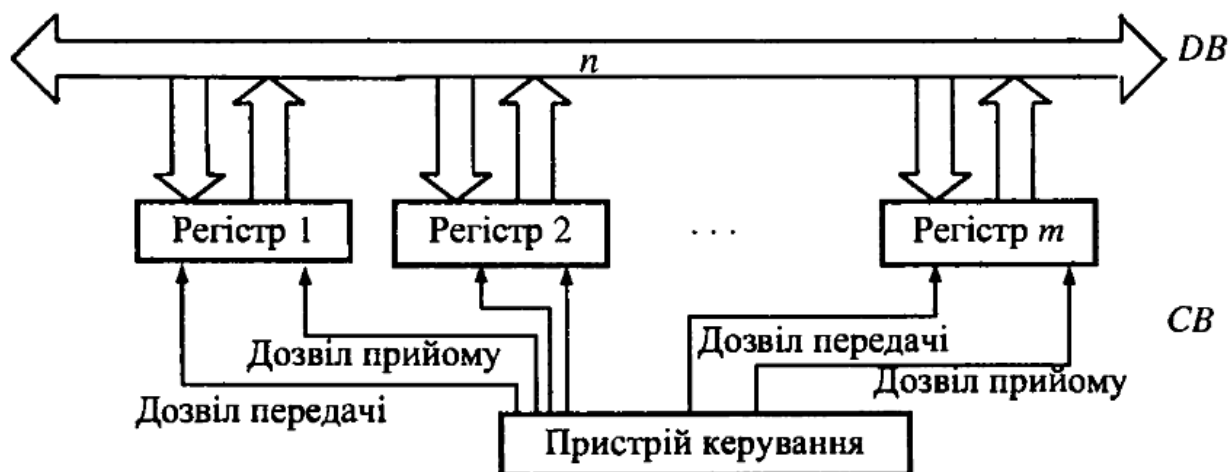
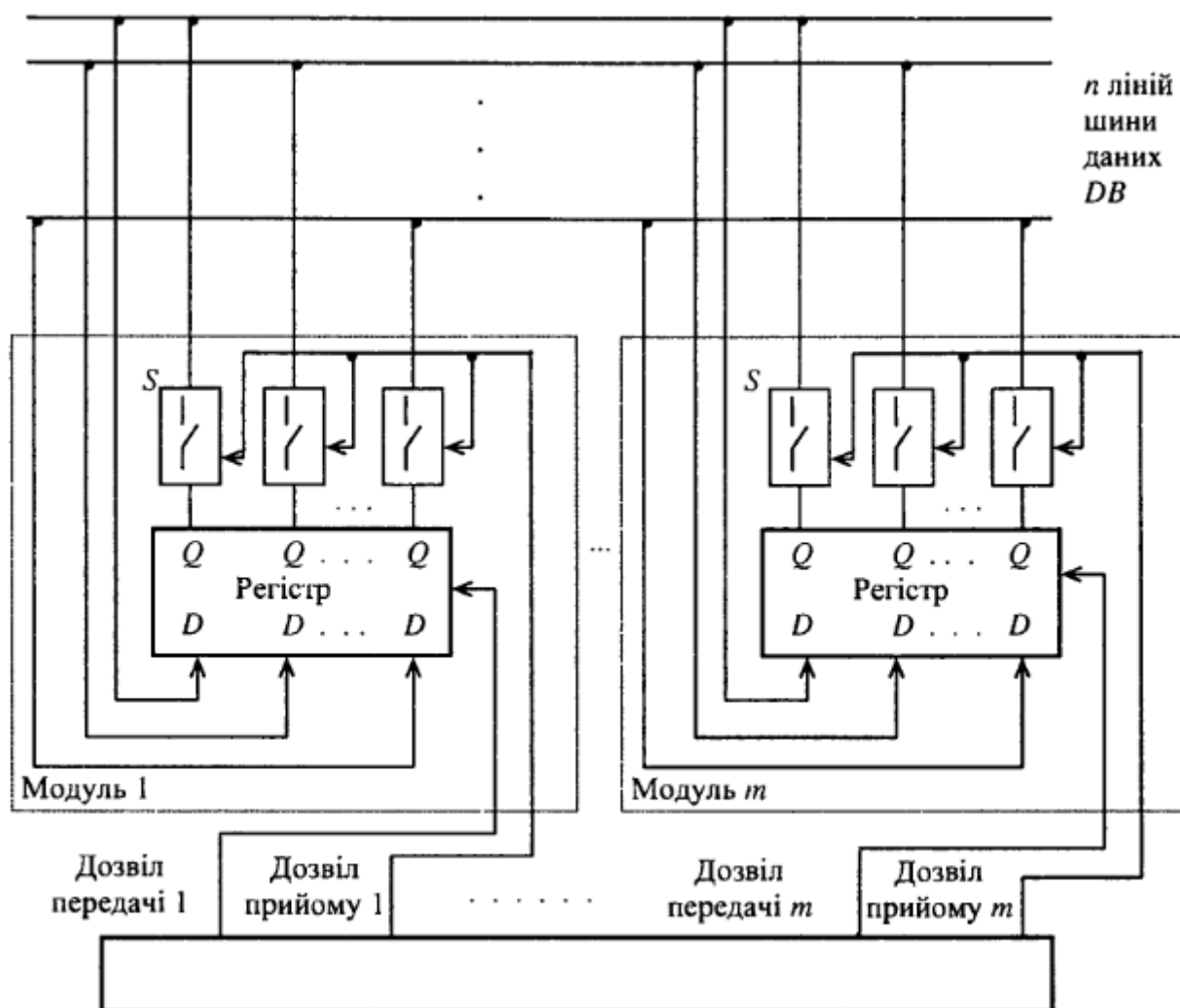


Рис. 7.3. Структурна схема передавання інформації між  $m$  регістрами по внутрішній « $n$ -розрядній шині даних

Дані по шині з  $n$  ліній передаються в режимі розподілу часу. Пристрій керування в кожний момент часу визначає адресу регістра, який передає інформацію, та регістра, який приймає інформацію. Для цього пристрій керування генерує сигнали *Дозвіл передачі* і *Дозвіл прийому*, що передаються по лініях шини керування. Лінії шини і сигнали керування мають назви *Дозвіл передачі* і *Дозвіл прийому*. У кожний момент часу передавати інформацію в шину може тільки один регістр. Це означає, що у разі надходження сигналу *Дозвіл передачі* до шини приєднується тільки один модуль (рис. 7.4).

Вхідні лінії регістрів з'єднані безпосередньо з відповідними лініями шини. Тому під час подання сигналу *Дозвіл прийому*, який надходить по

окремій лінії для кожного регістра, дані передаються по шині у відповідний регістр. Вихідні лінії регістрів з'єднуються з відповідними лініями шини через ключі  $5^i$ , що допускають монтажну логіку. Сигнал *Дозвіл передачі* надходить на ключі від пристрою керування по окремій для кожного регістра лінії.



. Рис. 7.4. Розширена структурна схема передавання інформації між  $m$  регістрами по внутрішній  $n$ -розрядній шині даних

Модуль, який братиме участь в обміні інформацією, визначається за одним з таких способів: 1) через відповідні лінії шини керування, окремі для кожного модуля (див. рис. 7.4); 2) за допомогою  $k$  ( $k = \log_2 m$ ) ліній шини адреси, по яких передається ідентифікаційний код, що ставиться у відповідність кожному модулю й однозначно його визначає; 3) використанням одних і тих самих ліній шини даних для передавання адрес і даних.

Дані по шині можуть передаватися у двох режимах: синхронному й асинхронному. У синхронному режимі пристрій керування визначає модулі, що беруть участь в обміні інформацією, синхронізує роботу

модулів та керує процесом обміну, виробляючи відповідні сигнали керування і синхронізації.

В асинхронному режимі модулі, готові до обміну, ініціюють процес передавання та прийняття інформації, виробляючи відповідні сигнали готовності.

## 7.2. Принципи побудови мікропроцесорних систем

В основу побудови МПС систем покладено три принципи: магістральності; модульності; мікропрограмного керування.

*Принцип магістральності* визначає характер зв'язків між функціональними блоками МПС – усі блоки з'єднуються з єдиною системною шиною.

*Принцип модульності* полягає в тому, що система будується на основі обмеженої кількості типів конструктивно і функціонально завершених модулів. Кожний модуль МПС системи має вхід керування третім (високоімпедансним) станом. Цей вхід називається *CS (Chip Select)* – вибір кристала або *OE (Output Enable)* - дозвіл виходу.

Дію сигналу *CS* для тригера показано на рис. 7.5. Вихідний сигнал тригера *Q* з'явиться на виводі лише при активному (у цьому випадку – нульовому) рівні сигналу *CS*. Якщо  $CS=1$ , тригер переводиться у високоімпедансний стан. Вихід тригера є тристабільним, тобто може знаходитися в одному з трьох станів: логічної одиниці, логічного нуля або у високоімпедансному. У кожний момент часу до системної шини МПС приєднано лише два модулі – той, що приймає, і той, що передає інформацію. Інші знаходяться у високоімпедансному стані.

Принципи магістральності та модульності дозволяють нарощувати керувальні й обчислювальні можливості МП через приєднання інших модулів.

*Принцип мікропрограмного керування* полягає у можливості здійснення елементарних операцій – мікрокоманд (зсуву, пересилання інформації, логічних операцій). Певною комбінацією мікрокоманд можна створити набір команд, який максимально відповідатиме призначенню системи, тобто створити технологічну мову. У секційних процесорах набір мікрокоманд можна змінити, використовуючи інші мікросхеми пам'яті мікрокоманд.

Узагальнену структурну схему МПС показано на рис. 7.6. До складу МПС входять: центральний процесор (ЦП), ПЗП, ОЗП; система переривань, таймер, ПВВ. Пристрої введення-виведення приєднані до системної шини через інтерфейси введення-виведення.

Постійний та оперативний запам'ятовувальні пристрої складають систему пам'яті, призначену для збереження інформації у вигляді

двійкових чисел. Постійний запам'ятовувальний пристрій призначений для збереження програм керування таблиць, констант, ОЗП – для збереження проміжних результатів обчислень. Пам'ять організовано у вигляді масиву комірок, кожна з яких має свою адресу і містить байт або слово. Байтом називається група із 8 біт, а слово може мати будь-яку довжину в бітах. Найчастіше під словом розуміють двійкове число довжиною 2 байти.

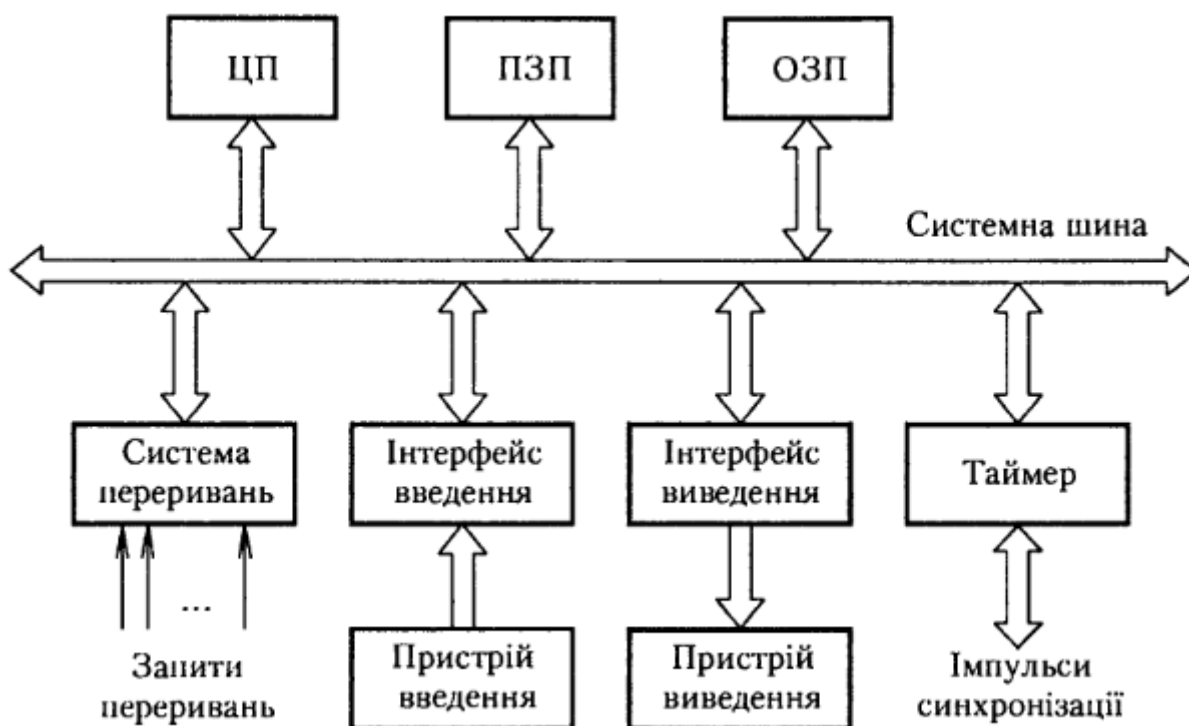


Рис. 7.6. Узагальнена структурна схема мікропроцесорної системи керування

Для звернення до комірки пам'яті треба видати її адресу на шину адреси. На рис. 7.7 зображено структуру пам'яті з 8 однобайтових комірок, де кожній адресі відповідає певний вміст комірки. Так, комірка з адресою 000 має вміст  $01011111_2 = 5F_{16}$ .

Адреса	Дані
000	01011111
001	00010011
010	01110111
011	00001100
100	00000000
101	11111111
110	10101010
111	11110000

Рис 7.7. Структура пам'яті з 8 однобайтових комірок

*Сегментом* називається область пам'яті, яка складається з суміжних комірок. У МП /8086 сегмент завжди починається з адреси, кратної 16, і займає до 64 кбайт. Існують три основні сегменти: кодів, даних, стека.

Сегмент кодів містить коди команд, які адресуються сегментним регістром кодів *CS* та регістром - вказівником команд *IP*. Регістр *CS* визначає початкову адресу сегмента кодів, а регістр *IP* - зміщення в сегменті (відстань від початку сегмента до комірки, в якій знаходиться адреса команди).

Сегмент даних містить дані, константи та робочі області, потрібні для виконання програми. Регістр *DS* має початкову адресу сегмента даних, а зміщення в сегменті задається в команді.

Сегмент стека містить адреси повернення з підпрограм та дані. Регістр *SS* має початкову адресу сегмента стека, а регістр *SP* - зміщення в сегменті.

Деякі операції використовують додатковий сегмент даних, початкова адреса якого задається регістром *ES*, а зміщення в сегменті визначається командою.

Двобайтове зміщення (16 біт) може знаходитися в межах від  $0000_{16}$  до  $FFFF_{16}$ . Для звернення до будь-якої адреси у програмі виконується додавання адрес, які знаходяться в регістрі сегмента та зміщення. Наприклад, перший байт у сегменті кодів має зміщення нуль, другий байт – одиницю і так далі до  $FFFF_{16}$ .

Конкретна адреса команди (для сегмента кодів), комірки пам'яті (для сегмента даних та додаткового сегмента) або комірки стека (для сегмента стека) визначається результатом додавання адреси сегмента, яка міститься у відповідному сегментному регістрі, та зміщення.

Модуль центрального процесора обробляє дані та керує всіма іншими модулями системи. Центральний процесор, крім ВІС МП, містить схеми синхронізації та інтерфейсу із системною шиною. Він вибирає коди команд з пам'яті, дешифрує їх і виконує. Протягом часу виконання команди – командного циклу ЦП виконує такі дії:

- виставляє адресу команди на шину адреси *AB*;
- отримує код команди з пам'яті та дешифрує його;
- обчислює адреси операнда і зчитує дані;
- виконує операцію, визначену командою;
- сприймає зовнішні керувальні сигнали, (наприклад, запити переривань);
- генерує сигнали стану і керування, потрібні для роботи пам'яті та ПВВ.

Пристрої введення-виведення або зовнішні пристрої – це пристрої, призначені для введення інформації у МП або виведення інформації з нього. Прикладами ПВВ є дисплеї, друкувальні пристрої, клавіатура, цифро-аналоговий та аналого-цифровий пристрої, реле, комутатори. Для з'єднання ПВВ із системною шиною їх сигнали мають відповідати певним стандартам. Це досягається за допомогою інтерфейсів введення-виведення.

Інтерфейси введення-виведення виконують функцію узгодження сигналів ПВВ із сигналами системної шини МП. Їх називають також контролерами або адаптерами. Мікропроцесор звертається до інтерфейсів за допомогою спеціальних команд введення-виведення. При цьому МП виставляє на шину адреси *AB* адресу інтерфейсу, а по шині даних *IB* зчитує дані з пристрою введення або записує у пристрій виведення.

Система переривань дозволяє МПС реагувати на зовнішні сигнали - запити переривань, джерелами яких можуть бути: сигнали готовності від зовнішніх пристроїв, сигнали від генераторів, сигнали з виходів датчиків. Із появою запиту переривання ЦП перериває основну програму і переходить до виконання підпрограми обслуговування запиту переривання. Для побудови системи переривань МПК містять В1С спеціальних програмованих контролерів переривань.

Таймер призначений для реалізації функцій, пов'язаних з відліком часу. Після того, як МП завантажує в таймер число, яке задає частоту, затримку або коефіцієнт ділення, таймер реалізує потрібну функцію самостійно.

### **7.3. Архітектура мікропроцесорів**

Поняття архітектури мікропроцесора визначає його складові частини, а також зв'язки та взаємодію між ними. Архітектура містить: 1) структурну схему самого МП; 2) програмну модель МП (опис функцій регістрів); 3) інформацію про організацію пам'яті (ємність пам'яті та способи її адресації); 4) опис організації процедур введення-виведення.

Існують два основні типи архітектури - фоннейманівська та гарвардська. *Фоннейманівську архітектуру* (рис. 7.8, а) запропонував 1945 року американський математик Джо фон Нейман.

Особливістю цієї архітектури є те, що програма і дані знаходяться у спільній пам'яті, доступ до якої здійснюється по одній шині даних і команд.

*Гарвардську архітектуру* вперше реалізовано 1944 року в релейній обчислювальній машині Гарвардського університету (США).

Особливістю цієї архітектури є те, що пам'ять даних і пам'ять програм розділені та мають окремі шини даних і шини команд (рис. 7.8, б), що дозволяє підвищити швидкодію МП системи.

Структурні схеми обох архітектур містять: процесорний елемент, пам'ять, інтерфейси введення-виведення (ІВВ) і ПВВ. Пам'ять і ІВВ для різних типів МП можуть бути як внутрішніми, тобто розміщуватися на тому ж кристалі, що і процесорний елемент, так і зовнішніми. Процесорний елемент містить регістри, арифметико-логічний пристрій (АЛП), пристрій керування і виконує функції обробки даних та керування процесами обміну інформацією. Пам'ять забезпечує зберігання кодів команд програми і даних. Інтерфейси призначені для зв'язку з ПВВ (наприклад, з клавіатурою, дисплеєм, друкувальними пристроями, датчиками інформації). Усі елементи структурної схеми з'єднані за допомогою шин.

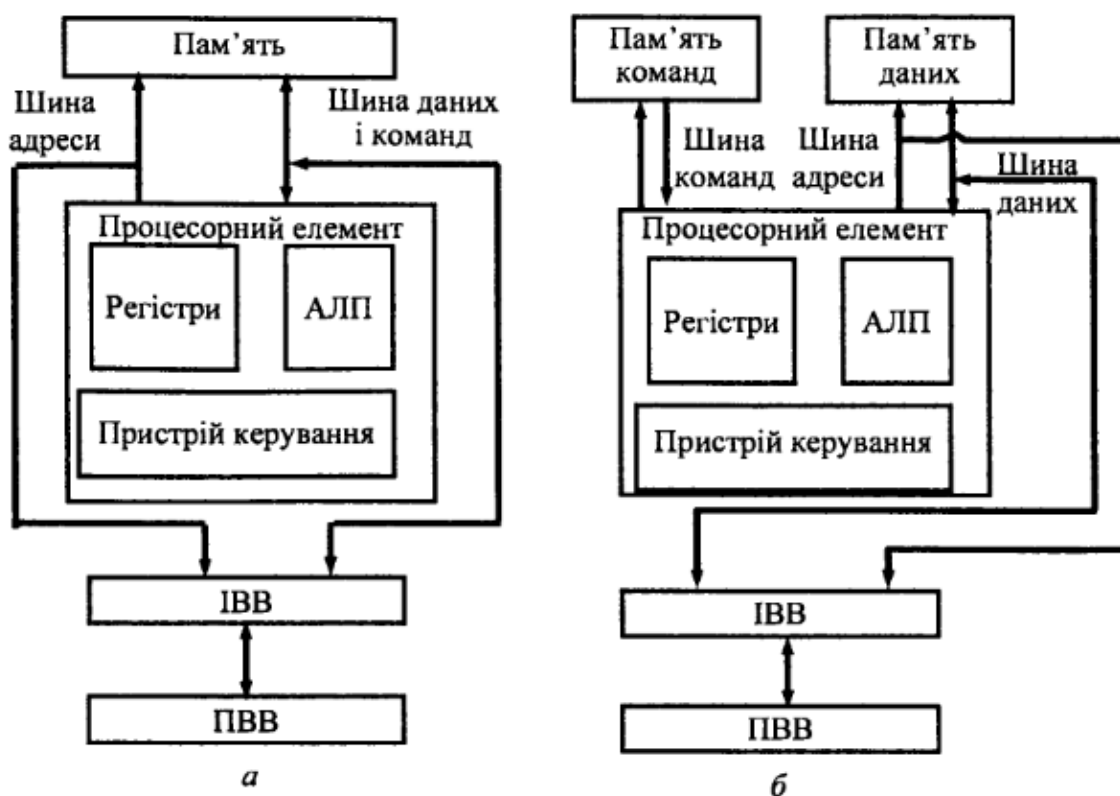


Рис. 7.8. Основні типи архітектури: а - фоннейманівська; б - гарвардська

Розширену структурну схему з процесором фоннейманівської архітектури показано на рис. 7.9.

Схема процесора містить пристрій керування, АЛП і регістри: адреси, даних, команд, стану, а також акумулятор, лічильник команд та вказівник стека.

*Пристрій керування* відповідно до кодів команд та зовнішніх керу- льних сигналів і сигналів синхронізації виробляє керувальні сигнали для всіх блоків структурної схеми МП, а також керує обміном



інформацією між МП, пам'яттю і ПБВ. Пристрій керування реалізує такі функції:

1. Функція початкового встановлення МП. Зовнішній сигнал початкового встановлення процесора *KE8ET* формується при ввімкненні джерела живлення МП або при натисканні кнопки *KE5ET*. У разі появи цього сигналу пристрій керування забезпечує завантаження нульового значення у програмний лічильник, що ініціює вибирання з пам'яті байта команди з нульовою адресою. Наприкінці вибирання вміст лічильника команд збільшується на одиницю і вибирається байт команд з наступною адресою. Таким чином виконується вся записана у пам'яті програма.

1. Функція синхронізації. Згідно із зовнішніми керувальними сигналами і сигналами синхросигналізації пристрій керування синхронізує роботу всіх блоків МП.

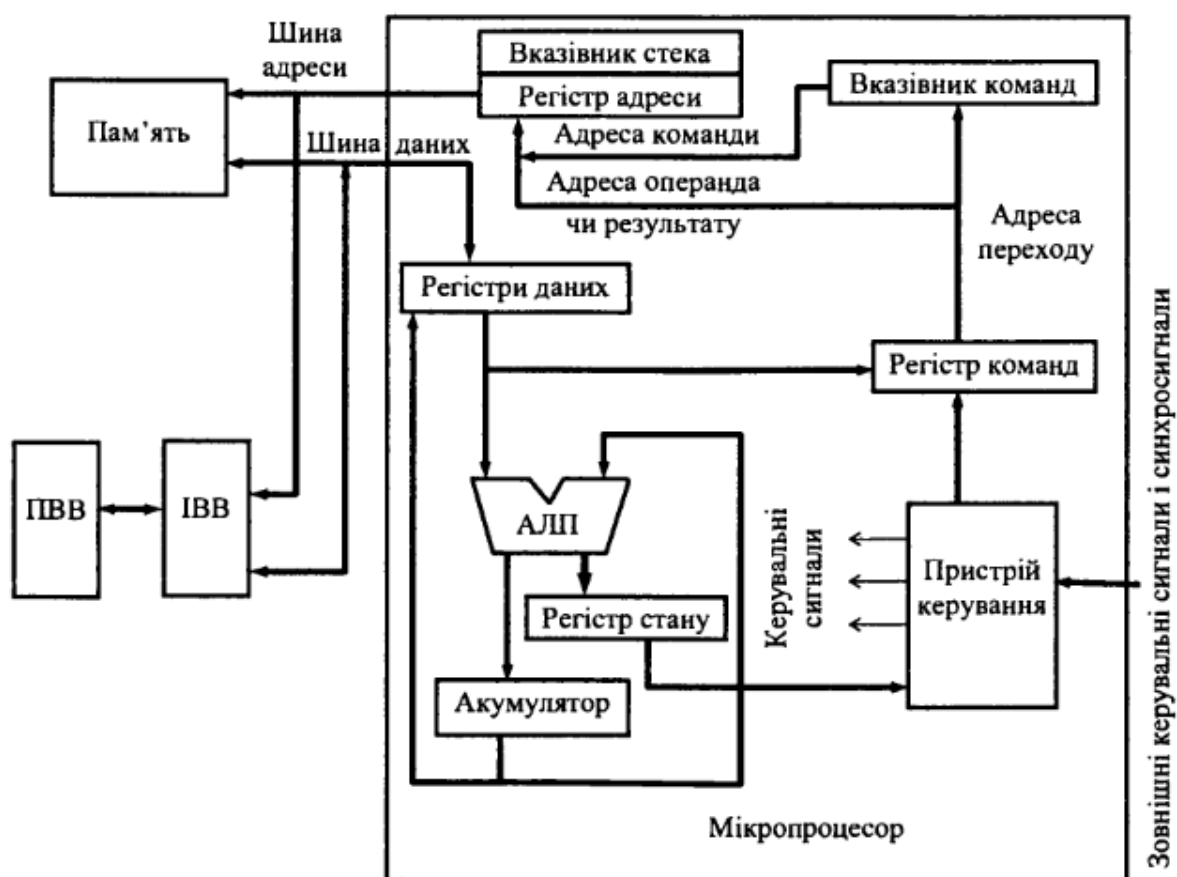


Рис. 7.9. Структурна схема з процесором фоннейманівської архітектури

3. Функція переривань. Із надходженням сигналу переривання пристрій керування ініціює роботу підпрограми обробки відповідного переривання. Потреба у реалізації функцій переривань виникає тоді, коли під час виконання основної програми треба перевести МП на розв'язання іншої задачі, наприклад, обробки аварійної ситуації або роботи з ПБВ.

4. Функція узгодження швидкодії модулів мікропроцесорної системи. Під час обслуговування пам'яті та ПВВ із значно меншою швидкодією, ніж МП, узгодження швидкодії вирішується генерацією тактів очікування МП, а під час обслуговування пристроїв з більшою швидкодією, ніж МП, використовується режим безпосереднього доступу до пам'яті.

*Арифметико-логічний пристрій* являє собою комбінаційну схему на основі суматора, який сигналами з виходів пристрою керування налагоджується на виконання певної арифметичної або логічної операції: додавання, віднімання, ЛОГІЧНЕ І, ЛОГІЧНЕ АБО, ЛОГІЧНЕ НІ, ВИКЛЮЧНОГО АБО, зсуву, порівняння, десяткової корекції. Отже, АЛП виконує арифметичні або логічні операції над операндами, які пересилаються з пам'яті і(або) регістрів МП.

Операнд – це об'єкт у вигляді значення даних, вмісту регістрів або вмісту комірки пам'яті, з яким оперує команда, наприклад, у команді додавання операндами є доданки. Операнд може задаватися у команді у вигляді числа або знаходитися в регістрі чи комірці пам'яті. Одержаний після виконання команди в АЛП результат пересилається в регістр або комірку пам'яті.

*Регістри* призначені для зберігання 2-розрядного двійкового числа. Вони являють собою п тригерів зі схемами керування читанням/записом та вибірки. Регістри створюють внутрішню пам'ять МП і використовуються для зберігання проміжних результатів обчислень.

*Акумулятор* – це регістр, у якому зберігається один з операндів. Після виконання команди в акумуляторі замість операнда розміщується результат операції. У 8-розрядних процесорах акумулятор бере участь в усіх операціях АЛП. У 16-розрядних МП більшість команд виконуються без участі акумулятора, але в деяких командах (введення, виведення, множення, ділення) акумулятор діє так само, як і у 8-розрядних МП, тобто зберігає один з операндів, а після виконання команди - результат операції.

*Вказівник команд, або програмний лічильник\**, призначений для зберігання адреси комірки пам'яті, яка містить код наступної команди. Програму дій МП записано в пам'яті у вигляді послідовності кодів команд. Для переходу до наступної команди вміст лічильника збільшується на одиницю у момент вибирання команди з пам'яті. Наприкінці виконання команди в лічильнику команд зберігається адреса наступної команди.

*Вказівник стека* - це регістр, який зберігає адресу останньої зайнятої комірки стека. Стеком або стековою пам'яттю називається

область пам'яті, організованої за принципом «останній прийшов - перший пішов».

Регістр команд зберігає код команди протягом усього часу виконання команди.

Регістр адреси і реєстри даних призначені для зберігання адрес і даних, використовуваних під час виконання поточної команди у МП.

Регістр стану або реєстр прапорців (ознак) призначений для зберігання інформації про результат операції в АЛП і являє собою декілька тригерів, які набувають одиничних або нульових значень. Наприклад, прапорець нуля встановлюється в одиницю при нульовому результаті операції.